

MineXpert User Manual

Deep mass spectrometric MS^n data visualization and exploration

Version: 12.4.0

Filippo Rusconi

Filippo Rusconi, 2026

Version 12.4.0
Published: 10/11/2026

To María Cecilia

To all the admirable people acting in the Free Software Movement *for a better and more ethical computing world*

To all involved in the development of the K Desktop Environment (KDE) *and of the* Plasma *desktop*

To all the readers who helped me with this manual.

Contents

2 Generalities 26

2.1 Citing the *MineXpert3* software 26

2.2 General concepts and terminologies 26

2.3 Acquiring Mass Data Along Time: To Profile or Not To Profile? 26

2.4 Mass Data Visualization: To Combine or Not To Combine? 27

2.5 Examples of Various Mass Spectral Data Integrations 29

2.5.1 TIC -> MZ integration 29

2.5.2 TIC -> DT integration 30

2.5.3 Using multiple threads during mass data integrations 31

2.6 Installation of the software 31

2.6.1 Installation on MS Windows 31

2.6.2 Installation on Debian- and Ubuntu-based systems 32

2.6.3 Installation with an AppImage software bundle 32

2.7 Building the software from source 33

2.7.1 The dependencies required to build *MineXpert3* 33

2.7.2 Getting the source tarball 34

2.7.2.1 Using Git 34

2.7.3 Building of the software 34

3 Configuring the application preferences 36

3.1 The application preferences window 36

3.2 Setting the general options 36

3.3 Setting the shortcuts 38

3.4 Setting the recording of the discoveries 38

4 The program's graphical user interface 40

4.1 Opening Mass Spectrometry Data Files 40

4.2 The Window Layout 42

4.3 The Main Program Window Menu 43

4.4 The Loaded MS Run Data Sets 44

4.5 Mass Spectral Data Listing in a Table View 45

4.6 The Main Data-Plotting Windows 48

4.6.1 The TIC Chromatogram Window 50

4.6.2 The Mass Spectrum Window 52

4.6.3 The Drift Spectrum Window 53

4.6.4 The Retention Time *vs* Mass Spectrum Color Map Window 54

4.6.5 The Drift Time *vs* Mass Spectrum Color Map Window 55

4.6.6 The Drift Time *vs* Retention Time Color Map Window 56
4.6.7 Exploring data in multiple plot widgets 57
4.7 General Structure of the Plot Widgets 59
4.7.1 The Left Column of Buttons Configures the Reception of New Data 60
4.7.2 The Right Column of Buttons Configures the New Integration to Run 62
4.7.3 The Composite Plot Widget Main Menu 64
4.7.3.1 The Graph|Bars Composite Plot Widget Main Menu 64
4.7.3.2 The Color Map Composite Plot Widget Main Menu 69
4.8 General Operation of the Plot Widgets 72
4.8.1 General Data Exploration Methods 72
4.8.2 Locking All the Composite Plot Widgets Views on X and/or Y Axes 74
4.8.3 Checking for Precise Alignment of Features Across Composite Plot Widgets 75
4.8.4 General Plot Management in the Composite Plot Widgets 75

5 Mass Data Integrations Featured by *MineXpert* 77
5.1 General Behaviour of Plot Widgets 77
5.1.1 Processing Flow Entities Document all Integrations 77
5.1.2 Setting the MS^n Fragmentation Parameters 79
5.1.3 Setting the *m/z* Integration Parameters 81
5.1.4 Step-by-step Determination of the Best *m/z* Integration Parameters 84
5.1.5 Removing 0-intensity *m/z* data points is useful 89
5.1.6 Savitzky-Golay Filtering of Any Kind of Data 89
5.2 Various Mass Spectral Data Integrations 91
5.2.1 Considerations on the Diversity of Mass Data Contents 94
5.2.2 Statistical Analysis of Mass Data 95
5.3 Integrations Originating in Color Map Plots 95
5.4 Signal Enhancement and Noise Reduction in Color Map Plots 98
5.5 Low Intensity Signal Removal in Traces 100
5.6 Chained Integrations 104
5.7 Multi-widget Intensity Integrations 107
5.8 Cumulative Intensity Integrations 108

6 Centroid Extraction from Mass Spectra 110

7 Mass Spectral Deconvolutions 114
7.1 Deconvolution based on charge state envelope inter-peak distance 115
7.2 Deconvolutions based on isotopic cluster peaks 117
7.2.1 Two-point charge determination with M_r calculation 117
7.2.2 Multi-point statistical analysis-based charge determination 119

7.2.3 Multi-point statistical analysis-based charge determination and mass estimations 127
7.3 Low Mass Automated Deconvolution Based on Fully Resolved Isotopic Clusters 129
7.4 High Mass Automated Deconvolution Based on Fully Resolved Isotopic Clusters 135
7.5 Reading the resolving power based on mass spectral data 140
8 Analyte identifications 142
9 The Processing Flow in a Data Exploration Session 147
9.1 Recording Processing Steps Linked to the Initial Mass Spectrometry Dataset 147
9.2 Recording Processing Steps Detached from the Initial Mass Spectrometry Dataset 153
10 Recording data exploration discoveries 156
Appendix 160
GNU General Public License version 3 160
Preamble 160
TERMS AND CONDITIONS 161
0. Definitions. 161
1. Source Code. 162
2. Basic Permissions. 163
3. Protecting User's Legal Rights From Anti-Circumvention Law. 163
4. Conveying Verbatim Copies. 164
5. Conveying Modified Source Versions. 164
6. Conveying Non-Source Forms. 165
7. Additional Terms. 166
8. Termination. 168
9. Acceptance Not Required for Having Copies. 168
10. Automatic Licensing of Downstream Recipients. 168
11. Patents. 169
12. No Surrender of Other's Freedom. 170
13. Use with the #acronym(GNU) Affero General Public License. 171
14. Revised Versions of this License. 171
15. Disclaimer of Warranty. 171
16. Limitation of Liability. 172
17. Interpretation of Sections 15 and 16. 172
END OF TERMS AND CONDITIONS 172
How to Apply These Terms to Your New Programs 172
Colophon 174
About the authors 174

Book preparation 174

Index 175

List of Figures

Figure 1 Adresses to send bug reports or questions to 22
Figure 2 Application preferences window 36
Figure 3 Application preferences - General page 37
Figure 4 Application preferences - Shortcuts page 38
Figure 5 Application preferences - Discoveries page 39
Figure 6 Probing the specific *timsTOF* MS runs 41
Figure 7 General View of the Graphical User Interface 43
Figure 8 The open MS run data sets 45
Figure 9 The table view-based scrutiny of one MS run data set 46
Figure 10 Mass spectral data set filtering using combined criteria 48
Figure 11 Resizing the plot widgets with the splitter 49
Figure 12 The total ion current (TIC) chromatogram window 51
Figure 13 The mass spectrum window 53
Figure 14 The drift spectrum window 54
Figure 15 The int = f(rt, *m/z*) colormap window 55
Figure 16 The int = f(dt, *m/z*) colormap window 56
Figure 17 The int = f(rt, dt) colormap window 57
Figure 18 Activation of the vertical rule 58
Figure 19 Using the vertical rule across all the plot widgets 59
Figure 20 Description of the various widgets that make a composite plot widget 60
Figure 21 Single TIC intensity value for any mass spectral data feature 63
Figure 22 Trace plot widget main menu 65
Figure 23 Submenu to configure the integrations plot widget main menu 66
Figure 24 Submenu to list the plots in the composite plot widget 66
Figure 25 Submenu to manage the lists of labels 67
Figure 26 Submenu to manage the export of the plot data points 67
Figure 27 Submenu to select filters 68
Figure 28 Fitting a Gaussian curve on an isotopic cluster 69
Figure 29 Colormap composite plot widget main menu 70
Figure 30 Submenus available to modify the z axis scale of intensities 70
Figure 31 Colormap plot widget main menu 71
Figure 32 Low-pass intensity filter configuration for colormap plots 72

Figure 33 Zooming-in operation 73
Figure 34 Plot window tool bar 74
Figure 35 The vertical rule marker spans the whole window 75
Figure 36 Plot management inside a composite plot widget 76
Figure 37 Setting the MS^n fragmentation parameters 79
Figure 38 Shortcut to the fragmentation MS level setting 81
Figure 39 Unusable combination spectrum without binning 82
Figure 40 The m/z integration parameters window 83
Figure 41 Determination of the best *m/z* integration parameters for integration to mass spectra 85
Figure 42 Estimating the resolving power of a mass spectral acquisition 87
Figure 43 Determination of the ion charge by using the isotopic cluster 87
Figure 44 Effects of the bin size on the colormap involving one *m/z* dimension 88
Figure 45 Removing 0-intensity data points 89
Figure 46 Menus to configure and run the Savitzky-Golay filter to smooth traces 90
Figure 47 Configuration of the Savitzky-Golay filter to smooth traces 90
Figure 48 Integrating mass data from a TIC chromatogram to a mass spectrum 91
Figure 49 Integrating mass data from a mass spectrum to a single TIC intensity value using the processing flow 92
Figure 50 Integrating mass data from a mass spectrum to a single TIC intensity value using the plot data points 93
Figure 51 Integrations from a colormap plot using a skewed selection rectangle (first step) 96
Figure 52 Integrations from a colormap plot using a skewed selection rectangle (last step) 97
Figure 53 Integrations from a colormap plot using a skewed selection rectangle (result) 98
Figure 54 Example of signal-to-noise ratio enhancement 100
Figure 55 Example of low intensity signal removal (starting trace) 101
Figure 56 Example of low intensity signal removal (configuration) 102
Figure 57 Example of low intensity signal removal (creation of a new trace) 103
Figure 58 Example of low intensity signal removal (display of the new trace) 104
Figure 59 Example of chained integrations, colormap where data exploration may start 105

Figure 60 Multiple chained integrations 106
Figure 61 Performing simultaneous integrations to intenstity on multiple composite plot widgets 107
Figure 62 Results from an integration to intensity on multiple composite plot widgets 108
Figure 63 Starting cumulative intensity integrations 109
Figure 64 Performing multiple intensity integrations 109
Figure 65 Configuration of the centroidation of mass spectra 111
Figure 66 Menu to be used to extract centroids from a trace 112
Figure 67 The successfully extracted centroids are displayed in the form of a bar plot 112
Figure 68 Mass spectrum plot widget-specific deconvolution menu 114
Figure 69 Charge state-based mass deconvolution 116
Figure 70 Isotopic cluster-based mass deconvolution 118
Figure 71 Zoomed-in isotopic cluster for charge determination 120
Figure 72 Triggering the isotopic cluster-based charge determination process 121
Figure 73 Isotopic cluster-based charge determination 122
Figure 74 Isotopic cluster-based charge determination 123
Figure 75 Isotopic cluster-based charge determination 125
Figure 76 Gaussian fit of the isotopic cluster's centroids 128
Figure 77 Low mass deconvolution of mass spectral data 130
Figure 78 Parameter set governing the low mass deconvolver 131
Figure 79 The synthetic mass spectrum (blue) mimicks the observed input mass spectrum only for successfully deconvoluted features 134
Figure 80 High mass deconvolution parameter settings 136
Figure 81 Input mass spectrum of a fluorescent protein 139
Figure 82 Deconvoluted mass spectrum of a protein 140
Figure 83 Calculation of the resolving power 141
Figure 84 Configuration of the analyte identification 142
Figure 85 Workflow leading from a mass spectrum to analyte identity suggestions 144
Figure 86 TIC chromatogram 148
Figure 87 Mass spectrum 149
Figure 88 Isotopic cluster of the 6+ ion 149
Figure 89 Ion mobility (drift) spectrum 150

Figure 90 Intensities are determined for the various ion mobility peaks 150
Figure 91 Retro-integrations from ion mobility spectrum peaks to TIC chromatograms 152
Figure 92 A centroids bars plot is no more logically linked to the initial mass dataset 154

Revision history

12.4.0 November 2026 fr

Update the documentation to describe the new high mass deconvolution feature

12.3.1 September 2026 fr

- Update the documentation to describe the new discrimination of ProcessingStep records in the ProcessingFlow making the difference between plots that have still a meaningful link to the initial mass spectrometric dataset and the plots (like those from calculations like Gaussian fit or centroid extraction) that have lost any meaningful link to the initial mass dataset. The implications are discussed.
- Started conversion of the user manual format from DocBook to Typst (thanks to fellow developer Olivier Langella for making me discover this typesetting tool).

12.2.0 August 2026 fr

Update the documentation to describe the new plot management in the composite plot widgets.

12.0.0 July 2026 fr

Update the documentation to describe the isotopic cluster-based charge determination and subsequently the mass estimations using menu-based selectable algorithms.

11.8.0 June 2026 fr

Update the documentation to describe the keyboard key-based selection of the integration destination (M for mass spectrum, D for drift spectrum, R for Tic/Xic).

11.7.1 June 2026 fr

Update the documentation to describe the cumulative intensity integrations.

11.7.0 June 2026 fr

- Update the documentation to describe the new isotopic cluster-based charge determination by statistical analysis of the m/z spacing in the isotopic cluster in the left-mouse-click-drag movement-spanned mass spectral region.
- Update the documentation to describe the new analyte identification process.

11.1.0 March 2026 fr

Update the documentation to describe the new centroid extraction and low mass deconvolver features.

11.0.0 January 2026 fr

Update the documentation to describe the new application preferences feature and the new vertical rule x-axis label in the plot windows.

10.2.0 November 2025 fr

Update the documentation to describe the new deconvolution processes.

10.1.1 November 2025 fr

Update the documentation about specifying the format of analysis session discoveries, in particular about the new feature that lets the user define the number of decimals to be used when printing out float values.

10.1.0 November 2025 fr

Full port of MineXp to the new XpMass project's libraries. Add new chapter on JavaScript features now in MineXp.

9.3.1 December 2023 fr

Added brief documentation of the new MS run probing feature designed to allow the user to configure the loading of timsTOF MS run data from Bruker.

8.5.8 October 2022 fr

Entirely reworked the user manual to make it more modern from the coding point of view. The users do not see any difference with the previous version.

8.4.0 June 2022 fr

Update the user manual to document the fact that it is now possible to set the bin size even when computing colormaps involving m/z as one of their dimension.

8.3.0 June 2022 fr

Update the user manual with the description of the new plot ranges history resetting feature.

8.2.0 31 January 2022 fr

Update the user manual with the description of the new trace low signal threshold removal filter.

8.1.0 21 April 2021 fr

Update the user manual with the description of the new colormap filters used to enhance the signal-to-noise ratio.

8.0.1 24 March 2021 fr

- Update the user manual with a more explicit data set to illustrate the new skewed selection polygon.
- Mention the bibliographic reference to the newly accepted article about MineXp2 in the Generalities chapter.

8.0.0 23 March 2021 fr

- Update the user manual with a section pertaining to the new skewed selection polygon. That feature was asked months ago by a user of MineXp2 and could finally be implemented these last weeks.
- Mention the bibliographic reference to the newly accepted article about MineXp2 in the Generalities chapter.

7.4.0 2 January 2021 fr

Happy new year! With the writing of the new version of the Processing Flow chapter to match the new integration logging mechanisms in MineXp2.

7.4.0 December 2020 fr

- This revision introduces updates to document the following new features.
- User-definable limitation of the number of threads usable for the combination computations (good request from a reviewer of the article).
- GUI shortcut to the MS level to be targeted for the next integration (good request from a reviewer of the article).
- Document the build and installation of the software (another good request from a reviewer).

7.0.1 October 2020 fr

Update of the user manual to document the new combine+ and combine- combinations features.

7.0.0 September 2020 fr

Update of the user manual to document new features, in particular the MS^n mass spectrometric data mining and new mass data plotting features.

6.0.0 January 2020 fr

Full rewrite of the user manual to document the new mineXpert2 software program. The new mineXpert2 version is a full rewrite of the previous version such that it can handle MS^n mass spectrometric data. Major features were included in the new version that are described in detail in this user manual.

5.8.2 February 2019 fr

Rework the IsoSpec-based isotopic clust calculations to better describe the three available processes. Also, describe the new FWHM calculation feature in the PeakShaperDlg.

5.8.0 End January 2019 fr

Add section to described the newly developed feature about isotopic cluster calculations based on the IsoSpec library.

5.7.0 Beginning January 2019 fr

Finished porting of the historical LaTeX-based documentation to the DocBook/DAPS/FOP publishing system. Please, see Colophon for details.

not_set September 2018 fr

Update the document to include new features and to make fixes. Insist on the fact that mineXpert reads mzML data files.

not_set March 2018 fr

Update the document to include new m/z integration features and scripting capabilities.

not_set December 2016 fr

Major rewriting of the document to incorporate all the new features. Large chapter on the scripting of mineXpert.

not_set December 2017 fr

Refactored document. First version documenting almost all of the current features of the software program.

not_set May 2017 fr

Refactored document. First version documenting almost all of the current features of the mineXpert software program.

not_set November 2016 fr

Resume writing, with new program name: mineXpert.

not_set September 2016 fr

Start of writing.

1 Preface

1.1 Software feature offerings and intended audience

This manual is about the *MsXpertSuite* mass spectrometric software suite, a software environment that contains two modules:

- *MassXpert*: a module that allows users to define brand new polymer chemistries. The polymer chemistry definitions are then used to model linear polymer sequences. Once modelled, a polymer sequence can undergo chemical reactions (enzymatic or chemical cleavages, gas-phase fragmentations...). The obtained results are a model of what a mass spectrum would look like if the modelled experiment had actually been carried over up to the mass spectrometry analysis.
- *MineXpert*: a module that allows users to load mass spectrometry data from mass spectrometry data files. The loaded data are then visualized and explored throughout all their MS^n depth. The mass data visualization starts either at the TIC chromatogram level or the MS data table view and deepens to the MS^n mass spectra, the drift spectra, the XIC chromatograms, various color maps relating *m/z* values with either retention times or drift times. A wide array of mass spectral integrations are available to ease the MS^n data finest scrutiny.

This manual is therefore intended for people willing to learn how to use the comprehensive *MsXpertSuite* software package.

Mass spectrometry has gained popularity across the past twenty years or so. Indeed, developments in polymer mass spectrometry have made this technique appropriate to accurately measure masses of polymers as heavy as many hundreds of kDa, and of any chemical type.

There are a number of utilities—sold by mass spectrometer constructors with their machines, usually as a marketing "plus"—that allow predicting/analyzing mass spectrometric data obtained on polymers. These programs are usually different from a constructor to another. Also, there are as many mass spectrometric data prediction/analysis computer programs as there are different polymer types. You will get a program for oligonucleotides, another one for proteins, maybe there is one program for saccharides, and so on. Thus, the biochemist/massist, for example, who happens to work on different biopolymer types will have to learn to use several different software packages. Also, if the software user does not own a mass spectrometer, chances are he will need to buy all these software packages.

The *MsXpertSuite* mass spectrometric software is designed to provide *free* solutions to all these problems by providing the following features:

- *MassXpert*:
  - Model *ex nihilo* polymer chemistry definitions (in the *XpertDef* module that is part of the *MassXpert* program);
  - Peform simple yet powerful mass computations to be made in a mass desktop calculator that is both polymer chemistry definition-aware and fully programmable (that's the *XpertCalc* module also part of the *MassXpert* program);
  - Edit polymer sequences on a polymer chemistry definition-specific basis, along with chemical reaction simulations, finely configured mass spectrometric computations... (all taking place in the *XpertEdit* module that is the main module of the *MassXpert* program);
  - Customize the way each monomer will show up graphically during the program operation (in the *XpertEdit* module);
  - Edit polymer sequences with immediate visualization of the mass changes elicited by the editing activity (in the *XpertEdit* module);
  - Open an unlimited number of polymer sequences at any given time and of any given polymer chemistry definition type (in the *XpertEdit* module);
- *MineXpert*:
  - Load mass spectometry data files in the mzML format, thanks to the excellent *libpwiz* library of ProteoWizard[1] fame; mass data file loading can be performed in *full-memory* mode (all the data read from file are stored in memory) for faster operations or in *streamed* mode when loading files largers than the available memory.
  - Display the data in powerful ways in a unified graphical user interface. The interface was designed to integrate all the most useful characteristics of the various proprietary environments known by the author, thanks to the excellent *libqcustomplot* library;[2]
  - Configure the way mass spectrometry data integrations are performed from a combination standpoint and optionally apply a Savitzky-Golay smoothing filter;
  - Configure the level of MS data (MS^n) for which the integrations are performed. Optionally filter data by precursor ion *m/z* values or precursor spectrum indices;

[1] proteowizard.sourceforge.net
[2] qcustomplot.com

- Perform data mining by performing data integrations in various ways;
- Ion mobility mass spectrometry data are supported with a int=f(*m/z*,dt) colormap plot calculation;
- A specific data integration mode allows easy mass spectral intensity determination for any feature (peak in a TIC chromatogram, a mass spectrum, a drift spectrum, any colormap region);

1.2 Feedback to the developers

We are always grateful to any constructive feedback from the users.

The MsXpertSuite software team might be contacted *via* the following addresses:

```
msxpertsuite@msxpertsuite.org = general mailing list about msXpertSuite
bug-reports@msxpertsuite.org  = report bugs found in msXpertSuite software
```

Figure 1: Adresses to send bug reports or questions to

Adresses to send bug reports or questions to.

1.3 Brief history of *MsXpertSuite*.

- *1998–2000*

  The name *MassXpert* comes from a project I started while I was a post-doctoral fellow at the École Polytechnique (Institut Européen de Chimie et Biologie, Université Bordeaux 1, Pessac, France);

  The *MassXpert* program was published in *Bioinformatics* (Rusconi, F. and Belghazi, M. *Desktop prediction/analysis of mass spectrometric data in proteomic projects by using massXpert. Bioinformatics*, 2002, 644–655.

  At that time, *MS-Windows* was at the *Windows NT 4.0* version and the next big release was going to be *you will see what you'll see*: *MS-Windows 2000*.

  When I tried *MassXpert* on that new version (one colleague had gotten it with a new machine), I discovered that my software would not run normally (the editor was broken). The Microsoft technical staff would advise to *buy a new version of the compiler environment and rebuild*. This was a no-go: I did not want to continue paying for using something I had already produced with legitimate software.

- *2001–2006*

  During fall 1999, I decided that I would stop using *Microsoft* products for my development. At the beginning of 2000 I started as a CNRS research staff in a new laboratory

and decided to start fresh: I switched to *GNU/Linux* (I never looked back). After some months of learning, I felt mature to start a new development project that would eventually become an official GNU package: *GNU polyxmass*.

The *GNU polyxmass* software, much more powerful than what the initial *MassXpert* software used to be, was published in *BMC Bioinformatics* in 2006 (Rusconi, F. *GNU polyxmass: a software framework for mass spectrometric simulations of linear (bio-)polymeric analytes).* BMC Bioinformatics, *225–*.

Following that publication I got a lot of feedback (very positive, in a way) along the lines: *—Hey, your software looks very interesting; only it's a pity we cannot use it because it runs on GNU/Linux, and we only use MS-Windows*.

- *2007–2016*

  In december 2006, I decided to make a full rewrite of *GNU polyxmass*. The software of which you are reading the user manual is the result of that rewrite. I decided to "recycle" the *MassXpert* name because this software is written in C++, as was the first *MassXpert* software. Also, because the first *MS-Windows*-based *MassXpert* project is not developped anymore; taking that name was kind of a "revival" which I enjoyed. However, the toolkit I used this time is not the Microsoft Foundation Classes (first *MassXpert* version) but the Trolltech *Qt* framework (see menu **Help⇨About Qt** in *MassXpert*).

  Coding with the *Qt* libraries has one big advantage: it allows the developer to code once and to compile on the three main platforms available today: *GNU/Linux*, *MacOSX*, *MS-Windows*. Another advantage is that the *Qt* libraries are wonderful software, technically and philosophically (Free Software).

  > **Note**
  >
  > I have abandoned supporting macOS since a number of years because the system is too painful to support.

  The rewritten software was published in 2009 (Rusconi, F. *massXpert 2: a cross-platform software environment for polymer chemistry modelling and simulation/analysis of mass spectrometric data.Bioinformatics*, 2009, 2741–2742.

- *2016–2019*

  In 2016, I started a new project about visualization of mass spectrometric data. The project developed pretty quickly, as we needed at the mass spectrometry facility a software that would allow to cope efficiently with ion mobility mass spectrometric experimental data. *MineXpert* was thus started.

To bundle both *MassXpert* and *MineXpert* in a single software suite, I bought the *MsXpertSuite* domain name http://msxpertsuite.org and created that new name.

- *2019–2021*

  The *MineXpert* software was published in 2019. The reviewers acknowledged that it was a useful piece of software but complained about its not handling the MS/MS data. Reference to cite: Filippo Rusconi. mineXpert: Biological Mass Spectrometry Data Visualization and Mining with Full JavaScript Ability. *J. Proteome Res.* 2019, 18, 5, 2254-2259.

In response to the reviewers and for my personal project that I had started a little earlier, I decided to fully rewrite *MineXpert* into *MineXpert*. The new version of the software had to support MS^n data.

- *2021–2022*

  The *MineXpert* software was published in 2019. The main new capability is now the full handling of MS^n data. This new feature required the full rewriting of the software internal data model. See Langella O., Rusconi F. (2021) mineXpert2: Full-Depth Visualization and Exploration of MS^n Mass Spectrometry Data. *J. Am. Soc. Mass Spectrom.*, 4 (32) 1138-1141.

- *2022–2025*

  A major work is now undertaken: the native handling of the timsTOF data from the Bruker instrument vendor. This kind of data contains ion mobility data.

- *2025–2026*

  Rewrite of major portions of the software to ease evolution of the code base, in particular with a strong enhancement of code reuse by putting large portions of code in the *libXpertMassCore* and *libXpertMassGui* libraries, for use by *MineXpert* and *MassXpert* .

- *2026–*

  Following the almost full rewrite, development of new features has resumed, all along regular feature requests by the users of the software. In particular, data processing features have been added to &MineXp;, such as a new automatic deconvolution of low mass analytes with analyte identification (with fully labeled analytes handling), peak labeling management in the plot widgets, new plot management inside a plot widget, network-based communication between *MineXpert* and the sister application *MassXpert* .

1.4 Program and Documentation Availability and License

The programs and all the documentation that are shipped along with the *MsXpertSuite* software suite are available at http://www.msxpertsuite.org. Most of the time, a new version is published as source, and as binary install packages for *MS-Windows* (64-bit systems only). No *GNU/Linux* binary packages are created outside of the autobuilder of the various distributions. As a Debian Developer, the author creates *Debian*[3] packages that are uploaded on the distribution servers. These packages are available using the system's software management infrastructure (like using the *Debian*'s `apt` command, for example, or the graphical application).

The software and all the documentation are all provided under the Free Software license *GNU General Public License, Version 3, or later, at your option*. For an in-depth study of the *Free Software* philosphy, I kindly urge the reader to visit http://www.gnu.org/philosophy.

[3] http://www.debian.org/

2 Generalities

In this chapter, I wish to introduce some general concepts around the *MineXpert3* program and the way data elements are named in this manual and in the program.

A mass spectrometry experiment generally involves monitoring the *m/z* value of analytes injected in the mass spectrometer along a certain time duration. The *m/z* value of each detected analyte is recorded along with the corresponding signal intensity, so that a mass spectrum is nothing but a series of (*m/z*,i) pairs recorded along the acquisition duration. All along the acquisition, the precise moment at which a given analyte is detected (and its (*m/z*,i) pair is recorded), is called the retention time of that analyte (rt). This retention time is not to be misunderstood as the drift time of that analyte in an ion mobility mass spectrometry experiment.

2.1 Citing the *MineXpert3* software

Please, cite the software using the following citation:

mineXpert2: Full-Depth Visualization and Exploration of MS^n Mass Spectrometry Data. Olivier Langella and Filippo Rusconi. *Journal of the American Society for Mass Spectrometry*.(2021) 32 (4): 1138–1141.

2.2 General concepts and terminologies

Most generally, the mass spectrometer acquires an important number of spectra in, say, one second. But all these spectra are *combined* together, and, on the surface, the massist only sees a "slow" acquisition of 1 spectrum per second. This apparent slow acquisition rate is configurable. At the time of writing, generally 1 spectrum per second is recorded on disk. So, say we record mass spectra for 5 minutes, we would have recorded (5*60) spectra.

2.3 Acquiring Mass Data Along Time: To Profile or Not To Profile?

As a mass spectrometry user, the reader of this manual certainly has used mass spectrometers where mass spectra are acquired and stored in different ways:

- Mass spectra are acquired and summed— the next to the previous— in such a manner that one is left, at the end of the acquisition, with a single spectrum of which the various peak intensities have been increasing all along the acquisition. Indeed, in this mode, each new spectrum is actually *"combined"* to the previously acquired ones. The result-

ing mass spectrum that is displayed on screen and that gets ultimately stored on disk is called a *combined spectrum*. This is typically the way MALDI-TOF mass spectrometers are used when manually acquiring data from samples deposited onto sample plates. We refer to this kind of acquisition as an "accumulation" mode acquisition;

- Mass spectra are acquired and stored on disk as a single file containing all the spectra, appended one after the other. There is no combination of the spectra: each time a new spectrum is displayed on screen, that spectrum is appended to the file.[4] This is typically the case when mass spectra are acquired all along a chromatography run and is generally called a *"profile"* mode acquisition. Note that this profile mode acquisition must not be mistaken as the profile mass peak type that negates the centroid mass peak type.

2.4 Mass Data Visualization: To Combine or Not To Combine?

In the previous section, we mentioned *spectrum combination* a number of times. What does that mean, that spectra are *"combined"* together into a single *"combined spectrum"*? Say we have 200 spectra that need to be combined together into a *single* spectrum that summatively represents the data of these 200 spectra.

First, a new spectrum would be allocated *(result spectrum)*, entirely empty at first. Then, the very first spectrum of the 200 spectra is literally copied into that result spectrum. At this point the combination occurs, according to an iterative process that has the following steps:

- Pick the next spectrum of the 200-spectra dataset;
  - Pick the first (m/z, i) pair of the currently iterated spectrum;
  - Look up in the *result spectrum* if a m/z value identical to the m/z value of the current (m/z, i) pair is already present;
  - If the m/z value is found, increment its intensity by the intensity of the (m/z, i) pair;
  - Else, if the m/z value is not found, add the current (m/z, i) pair to the result spectrum;

  -Iterate over all the remaining (m/z, i) pairs of the current spectrum and redo these steps.
- Iterate over all the 198 remaining spectra of the dataset and do the steps above for each single iterated spectrum.

[4] Although there certainly is spectrum combination going on in the guts of the software, because the system actually acquires much more spectra than is visible on screen and each newly displayed spectrum is actually the combination of many spectra acquired under the surface.

At the end of the two nested loops above, the combined spectrum is still a single spectrum that represents— summatively— all the 200 spectra. This whole process is very computing-intensive, in particular if:

- The *m/z* range is large: there are lots of points in each spectrum, which means that for each new (*m/z*,i) pair we need to iterate in the long list of *m/z* values that make the result spectrum;
- The resolving power of the mass spectrometer is high: there are many points per *m/z* range unit.

When a profile mode acquisition is performed, the user gets an innumerable number of distinct spectra, all appended to a single file. These unitary spectra are virtually unusable if an initial processing is not performed. This initial processing of the spectra is called *"total ion current chromatogram calculation"*.

What is it? Let's say that the user has performed a profile mode mass spectrometry acquisition on the eluate of a chromatography column. Now, imagine that the spectrometer stores the mass data at a rate of one spectrum per second and that the chromatography gradient develops over 45 min: there would be a total of (45 * 60) spectra in that file. The question is:

> — *"How can we provide the user with a data representation that might be both meaningful and useful to start exploring the data?"*

The conventional way of doing so is to load all the mass spectra and compute the *"total ion current chromatogram"* (the TIC chromatogram). The analogy with chromatography is evident: the TIC chromatogram is the same as the UV chromatogram unless optical density is not the physical property that is measured over time; instead, the amount of ions that are detected in the mass spectrometer is measured over time. That amount is actually the sum of the intensities of all the (*m/z*,i) pairs detected in each spectrum. When mass data are acquired during a chromatography run, often, the total ion current chromatogram mirrors (mimicks) the UV chromatogram[5]. For each retention time (rt) a TIC value is computed by summing the intensities of all the (*m/z*,i) pairs detected at that specific rt.

How is this total ion current chromatogram computed? This is an iterative process: from the first spectrum (retention time value 0 s), to the second spectrum (retention time value 1 s) up to the last spectrum (retention time 45 min), the program computes the sum of the intensities of all the spectrum's (*m/z*,i) pairs. That computation ends up with a map that relates each rt value with the corresponding TIC value. The TIC chromatogram is nothing but a plot of the TIC values as a function of rt values. In that sense, it is indeed a chromatogram.

[5] Unless eluted analytes do absorb UV light but do not either desorb/desolvate or ionize, or both.

MineXpert3 works exactly in this way. When mass spectrometry data are loaded from a file, the TIC chromatogram is computed and displayed. This TIC chromatogram serves as the basis for the mass data exploration, as described in this manual. The TIC chromatogram serves as the basis for spectral combinations that can be performed in various ways—and not all formally *combinations*, which is why I prefer the term *"integrations"*. Some of these integrations are described below:

- Integrating data from the TIC chromatogram to a single mass spectrum;
- Integrating data from the TIC chromatogram to a single drift spectrum;

Note that the reverse actions are possible (and indeed necessary for a thorough data exploration): selecting a region of a mass spectrum and asking that the TIC chromatogram be reconstituted from there; or selecting a region of a drift spectrum and asking that the TIC chromatogram be reconstituted from there also. Finally, integrations may, of course, be performed from a mass spectrum to a drift spectrum, and reverse.

2.5 Examples of Various Mass Spectral Data Integrations

In the sections below, the inner workings of *MineXpert3* are described for some exemplary mass data integrations. For example, when doing ion mobility mass spectrometry data exploration, it is essential to be able to characterize most finely the drift time of each and any analyte. Since each analyte is actually defined as one or more (m/z, i) pairs, it is essential to be able to ask questions like the following:

- What is the drift time of the ions below this mass peak?
- What are all the drift times of all the analytes going through the mobility cell for a given retention time range?
- What are all the ions that are responsible for the shoulder right of a given ion mobility peak in the drift spectrum?

2.5.1 TIC -> MZ integration

What computation does actually *MineXpert3* do when a mass spectrum is computed starting from a TIC chromatogram region, say between retention time rt minute 7 and rt minute 8.5?

1. List all the mass spectra that were acquired between rt 7 and rt 8.5. In this *spectral set*, there might be many hundreds of spectra that match this criterion, if we think that, in ion mobility mass spectrometry, #asymp 200 spectra are acquired and stored individually every second (I mean it, every 1 s time lapse);

2. Allocate a new empty spectrum— the *"combined spectrum"*— and copy into it without modification the first spectrum of the spectral set;

3. Go to the next spectrum of the spectral set and iterate into each (*m/z*,i) pair:

   - Check if the *m/z* value of the iterated pair is already present in the combined spectrum. If so, increment the combined spectrum's (*m/z*,i) pair's intensity value by the intensity of the iterated (*m/z*,i) pair's intensity. If not, simply copy the iterated (*m/z*,i) pair into the combined spectrum;

   - Iterate over all the remaining (*m/z*,i) pairs and perform the same action as above.

+Iterate over all the remaining spectra of the spectral set and perform the steps for each spectrum.

MineXpert3 then displays the combined spectrum.

2.5.2 TIC -> DT integration

What computation does *MineXpert3* actually do when a drift spectrum is computed starting from a given TIC chromatogram region, say between retention time rt minute 7 and rt minute 8.5?

What is a drift spectrum? A drift spectrum (mobilogram) is a plot where the cumulated ion current of the detected ions is plotted against the drift time at which they were detected. Let's see how that computation is handled in *MineXpert3*:

1. Create a map to store all the (drift time, intensity) pairs that are to be computed below, the (dt,i) map;

2. List all the mass spectra that were acquired between rt 7 and rt 8.5. The obtained list of mass spectra is called the *"spectral set"*;

3. Go to the first spectrum of the spectral set and compute its TIC value (sum of all the intensities of all the (*m/z*,i) pairs of that spectrum). Get the drift time value at which this mass spectrum was acquired. We thus have a value pair: (dt, i), that is, for drift time *dt*, the intensity of the total ion current is *i*

   At this point, we need to do a short digression: we saw earlier that, at the time of this writing, one of the commercial instruments on which the author of these lines does his experiments stores 200 spectra each second. These 200 spectra actually correspond to the way the drift cycle is divided into 200 bins (time bins). That means that in the retention time range [7– 8.5], there are (1.5*60) complete drift cycles. And thus there are (1.5*60) spectra with drift time x, the same amount of spectra with drift time y, and so on for the reminaing 198 time bins. Of course, a large number of these spectra might be almost empty, but these spectra are there and we need to cope with them.

The paragraph above must thus lead to one interrogation about the current (dt,i) pair: —*"Has the current dt value be seen before, during the previous iterations in this loop?"*. If not, then create the (dt, i) pair and add it to the (dt,i) map; if yes, get the *dt* element in the map and increment its intensity value by the TIC value computed above;

4. Iterate over all the remaining spectra of the spectral set and perform the steps for each spectrum.

At the end of the loop above, we get a map in which each item relates a given drift time with a TIC value. This can be understood this way: —*"For each drift time value, what is the accumulated ion current of all the ions having that specific drift time?"*.

At this point, *MineXpert3* displays the drift spectrum (ion mobility spectrum, or mobilogram).

2.5.3 Using multiple threads during mass data integrations

Whenever the computer has multiple threads available for the integrations to be performed using parallel execution, *MineXpert3* manages to use all these available threads. It is possible to limit the number of threads used for the integration computations as described in the **Preferences⇨Set max. thread count** menu at Section 4.3 – The Main Program Window Menu.

2.6 Installation of the software

The installation material is available at msxpertsuite.org.

2.6.1 Installation on MS Windows

The installation of the software is extremely easy on the *MS-Windows* platforms. The installation program are standard and require no explanation.

> **Note**
>
> I have abandoned building *macOS* packages because of the volatility of the platform with system upgrades that impose rebuilding the software every now and then.

2.6.2 Installation on Debian- and Ubuntu-based systems

The installation on Debian- and Ubuntu-based GNU/Linux platforms is also extremely easy (even more than in the above situations). *MineXpert3* is indeed packaged and released in the official distribution repositories of these distributions and the only command to run to install it is:

`sudo apt install<package_name>` RETURN

In the command above, the typical <*package_name*> is in the form `minexpert3` for the program package and `minexpert3-doc` for the user manual package.

Once the package has been installed, the program shows up in the *Science* menu. It can also be launched from the shell using the following command:

`minexpert3` RETURN

> **Tip**
>
> If the Debian system onto which the program is to be installed is older than *testing*, that is, older than *Buster (Debian 10)*, then using the AppImage program bundle might be a solution. See below for the method to run *MineXpert3* as an AppImage bundle.

2.6.3 Installation with an AppImage software bundle

The *AppImage* software bundle format is a format that allows one to easily run a software program on any GNU/Linux-based distribution. From the appimage.org:

> The key idea of the AppImage format is *"one app = one file"*. Every AppImage contains an app and all the files the app needs to run. In other words, each AppImage has no dependencies other than what is included in the targeted base operating system(s).
>
> — Simon Peter

There are AppImage software bundles for the various *MineXpert3* versions that are available for download. As of writing, the software bundle has been tested on *Fedora* version 39 (release in 2023). This is thus a pretty old distribution and thus *MineXpert3* should also run on more recent versions of these computing platforms. The AppImage bundle of *MineXpert3* was created on a rather current Debian version: the *stable Debian* 13 distribution.

In order to run the *MineXpert3* software AppImage bundle, download the latest version (like `MineXpert3-\<version\>-x86_64.AppImage`. Once the file has been downloaded to the desired directory, change to that directory and change the permissions to make it executable:

```
chmod a+x MineXpert3<version>-x86_64.AppImage RETURN
```

Finally, execute the file that has become a normal program:

```
./MineXpert3-<version>-x86_64.AppImage RETURN
```



Tip

If the program complains about a locale not being found, please, modify the command line to read:

```
LC_ALL=C./MineXpert3-<version>-x86_64.AppImage
RETURN
```

2.7 Building the software from source

The *MineXpert3* software build is under the control of the *CMake* build system. There are a number of dependencies to install prior to trying to build the software, as described below.

2.7.1 The dependencies required to build *MineXpert3*

The dependencies to be installed are listed here with package names matching the packages that are in *Debian/Ubuntu*. In other RPM-based software, most often the package names are similar, albeit with some slight differences.

- **The build system**: cmake
- **Conversion of `svg` files to `png` files**: graphicsmagick-imagemagick-compat
- **For the parallel computations**: libgompı
- **For the isotopic cluster calculations**, libisospec++-dev
- **For low-level chemistry classes and high-level GUI classes**, libXpertMassCore-dev libXpertMassGui-dev
- **For all the raw mass calculations like the data model, the mass spectral combinations...**: libpappsomspp-dev, libpappsomspp-widget-dev
- **For all the plotting**: libqcustomplot-dev

- **For the C++ objects (GUI and non-GUI)**: qtbase5-dev, libqt5svg5-dev, qttools5-dev-tools, qtchooser
- **For the man page**: docbook-to-man
- **For the documentation (optional, with `-DMAKE_USER_MANUAL=1` as a flag to the call of *cmake*, see below.)**: typst

> **Typst is not in Debian**
>
> The Typst software is not yet packaged for Debian, which means it has been used locally and documentation in Debian packages is provided pre-built, along with the source code.

2.7.2 Getting the source tarball

The source tarball is available for the following internation Git repositories:

- Codeberg
- GitLab

In both websites above, select the sub-project and search for the download menu right next to the top of the file system view of the code (ellipsis or hamburger menu). Choose the format of the archive to be downloaded.

2.7.2.1 Using Git

The rather convoluted command below only downloads the branch of interest. The whole Git repos is very large...

```
git clone --branchmaster/7.3.0-1 --single-branch minexpert3-7.3.0
```

2.7.3 Building of the software

Change directory:

cd minexpert3-7.3.0

Create a build directory:

mkdir build

Change directory:

cd build

Configure the build:

cmake ../ -DCMAKE_BUILD_TYPE=Release

Build the software:

make

3 Configuring the application preferences

A large application like *MineXpert* has lots of configurable options. The Application preferences window described in this chapter is the central place for these customizations. To enter the application preferences window, select **Preferences⇨Edit preferences**, which opens the window at Figure 2 – Application preferences window.

3.1 The application preferences window

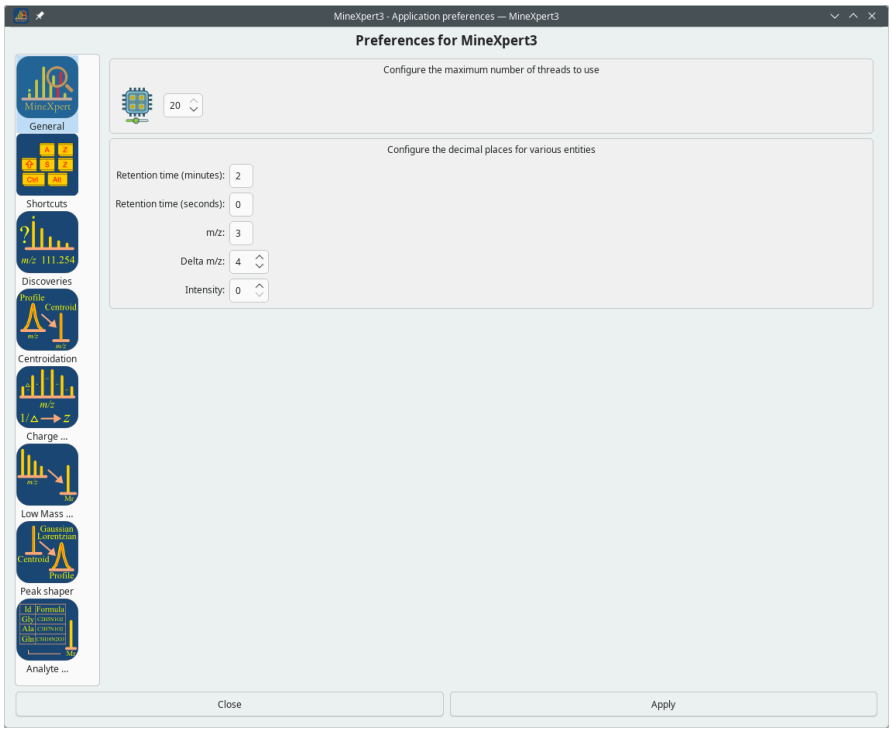


Figure 2: Application preferences window

The Application preferences window harbors a large set of preferences organized into sections, visible on the left hand side margin of the window.

3.2 Setting the general options

As shown in Figure 3 – Application preferences - General page the general options encompass the following notions:

- **Maximum number of threads:** Set the maximum number of threads when performing resource intensive tasks, like mass spectrum integrations.



Tip

Do not mistake “thread” for “core” or “processor”. When processors are described, the description usually says “x core”. That means that there are x “processors” in the whole central processing unit attached to the motherboard. Each processor most often can branch two different threads. So a computer that sports 2 cores, will most often in fact make 4 threads available for the computations.



Warning

It is of the utmost importance to make sure that guest virtualized environments do share at least 2 (better 4) threads with the host system. In *VirtualBox*, for example, it is possible to allocate a defined number of threads to the guest operating system. Ensure that the number of threads made available to the guest operating system is at least 2, with 4 being a right low level value.

- **Decimal places for number representation:** Set the numer of decimal digits when representing numbers. Note that not all numbers need the same decimal precision.

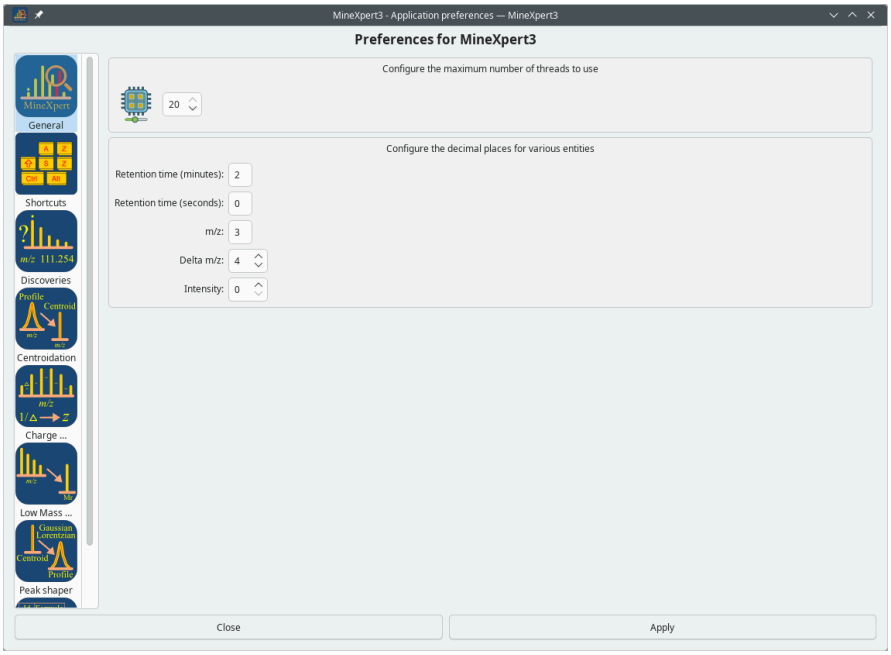


Figure 3: Application preferences - General page

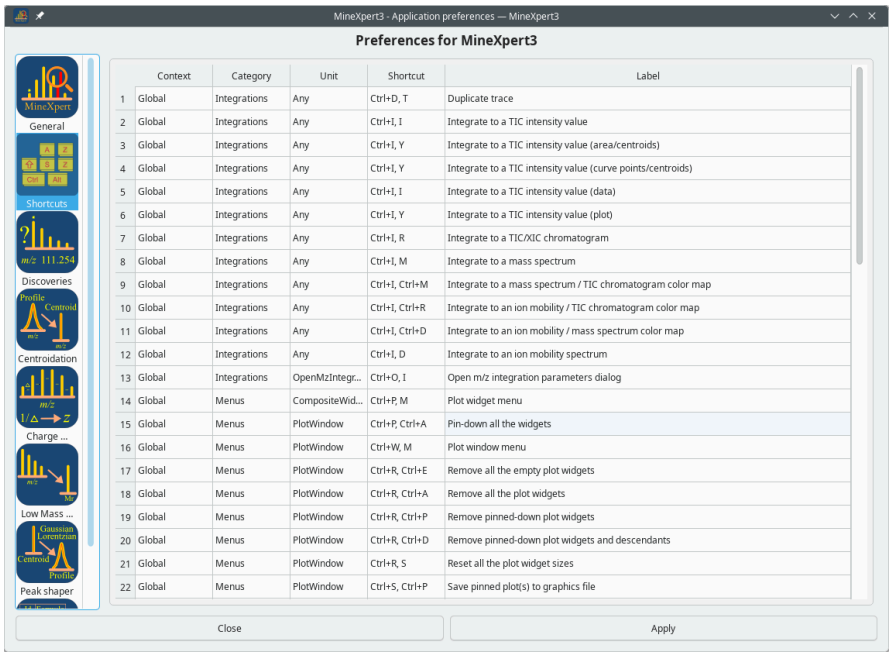
The General page configures the maximum number of threads to use and the number of decimals to be used for a number of numerical entities.

← *Configuring the application preferences*

The program's graphical user interface →

3.3 Setting the shortcuts

As shown in Figure 4 – Application preferences - Shortcuts page it is possible to configure the shortcuts for a large number of tasks throughout the application.



	Context	Category	Unit	Shortcut	Label
1	Global	Integrations	Any	Ctrl+D, T	Duplicate trace
2	Global	Integrations	Any	Ctrl+I, I	Integrate to a TIC intensity value
3	Global	Integrations	Any	Ctrl+I, Y	Integrate to a TIC intensity value (area/centroids)
4	Global	Integrations	Any	Ctrl+I, Y	Integrate to a TIC intensity value (curve points/centroids)
5	Global	Integrations	Any	Ctrl+I, I	Integrate to a TIC intensity value (data)
6	Global	Integrations	Any	Ctrl+I, Y	Integrate to a TIC intensity value (plot)
7	Global	Integrations	Any	Ctrl+I, R	Integrate to a TIC/XIC chromatogram
8	Global	Integrations	Any	Ctrl+I, M	Integrate to a mass spectrum
9	Global	Integrations	Any	Ctrl+I, Ctrl+M	Integrate to a mass spectrum / TIC chromatogram color map
10	Global	Integrations	Any	Ctrl+I, Ctrl+R	Integrate to an ion mobility / TIC chromatogram color map
11	Global	Integrations	Any	Ctrl+I, Ctrl+D	Integrate to an ion mobility / mass spectrum color map
12	Global	Integrations	Any	Ctrl+I, D	Integrate to an ion mobility spectrum
13	Global	Integrations	OpenMzIntegr...	Ctrl+O, I	Open m/z integration parameters dialog
14	Global	Menus	CompositeWid...	Ctrl+P, M	Plot widget menu
15	Global	Menus	PlotWindow	Ctrl+P, Ctrl+A	Pin-down all the widgets
16	Global	Menus	PlotWindow	Ctrl+W, M	Plot window menu
17	Global	Menus	PlotWindow	Ctrl+R, Ctrl+E	Remove all the empty plot widgets
18	Global	Menus	PlotWindow	Ctrl+R, Ctrl+A	Remove all the plot widgets
19	Global	Menus	PlotWindow	Ctrl+R, Ctrl+P	Remove pinned-down plot widgets
20	Global	Menus	PlotWindow	Ctrl+R, Ctrl+D	Remove pinned-down plot widgets and descendants
21	Global	Menus	PlotWindow	Ctrl+R, S	Reset all the plot widget sizes
22	Global	Menus	PlotWindow	Ctrl+S, Ctrl+P	Save pinned plot(s) to graphics file

Figure 4: Application preferences - Shortcuts page

The shortcuts are classified according to Context, Category and Unit.

The shortcuts can be configured by double-clicking the item of interest in the cell of the Shortcut column and then entering the desired shortcut. Then simply click somewhere else. When all the shortcuts have been configured, click the button labeled Apply and restart the program.

3.4 Setting the recording of the discoveries

As shown in Figure 5 – Application preferences - Discoveries page it is possible to configure the way discoveries are stored.

← *Configuring the application preferences*

The program's graphical user interface →

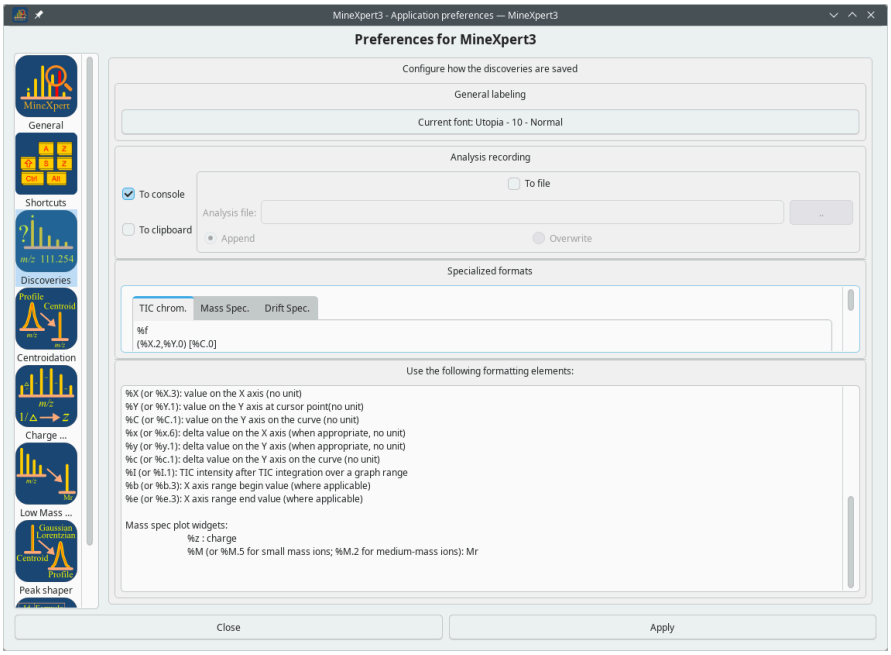


Figure 5: Application preferences - Discoveries page

The text format specification configures the output of the text describing the disovery (see text). That specification is defined for each one of the three main graphs: TIC/XIC chromatogram, mass spectrum and drift spectrum.

The detailed configuration of the discoveries storage is described in Section 10 – Recording data exploration discoveries.

4 The program's graphical user interface

Data exploration, in mass spectrometry, entails, for a large part, the relentless scrutiny of the mass spectra by an expert eye. Without a powerful mass spectrum viewer, capable of numerous data display modes, the expert eye remains powerless. In this chapter, the graphical user interface is described in detail, starting from the opening of a mass spectrometry data file, to the navigation through all the data depth within a large set of graphical representations of the data.

4.1 Opening Mass Spectrometry Data Files

To start a session, open one or more mass spectra using the menu **File⇨Open mass spectrum file(s) fully in memory**. *MineXpert* understands the mzML, mzXML and MGF formats. The file loading procedures, for these formats, are delegated to the excellent `libpwiz` library from the *ProteoWizard* project.[6] Simple txt, asc data where *m/z* and intensity values are separated by any character that is neither a newline nor a dot nor a digit (loading is handled by a private parser) can be loaded either from file or directly from the clipboard (menu **File⇨Load mass spectrum from clipboard**).

The successful *timsTOF* line of instruments from *Bruker* produces huge mass spectrometry data files. We provide the user with a mechanism to probe the data and configure the way they should be loaded. When *MineXpert* detects that such a file is being loaded, it first probes the complexity of the data in it and opens the dialog window shown in Figure 6 – Probing the specific *timsTOF* MS runs.

[6] Please, see http://proteowizard.sourceforge.net/.



Figure 6: Probing the specific *timsTOF* MS runs

The contents of the dialog window are self-explanatory.

There are two actions that can be triggered from the dialog window:

- Flatten IM scans: this option will compute one single mass spectrum for each retention time by combining together all the ion mobility spectra acquired at that time (there can be as many as a thousand ion mobility spectra).
- Keep IM scans: this option will keep all the ion mobility scans.

> **Note**
>
> The Keep IM scans option will force the software to load such a huge amount of spectra for a LC-MS/MS acquisition that this is going to be virtually impossible. The only way to actually load all the IM scans (spectra) is to limit the scope of the data loading using the various parameters configurable in the dialog window shown in Figure 6 – Probing the specific *timsTOF* MS runs.

There are two variants of the mass spectrometry file opening menu: one for which the mass data are read from file and stored *totally* in memory and one for which the mass data are read from file in *streamed mode*, used to compute the TIC chromatogram and discarded (menu **File⇨Load mass spectrum file in stream**). The latter mode is useful when the mass data are so large that they cannot fit in memory. Optionally, a table view can be displayed with all the data in the MS run data set that was loaded from disk. The TIC chromatogram or the table view can then be used to access the mass data in the file according to criteria set by the user (retention time range, for example).

4.2 The Window Layout

The graphical interface comprises a number of windows where data and informations are displayed. When first started, the program shows the main program window. Upon loading of a MS run data set from file, new windows do show up as described below (see Figure 7 – General View of the Graphical User Interface):

- *MineXpert* main program window: this is an unintrusive window sporting the main menu;
- The Task monitors window, that displays all the ongoing tasks, always providing the user the ability to cancel an ongoing task;
- The Open MS run data sets window, that lists all the mass spectrometry data sets that have been loaded from file;
- The TIC/XIC chromatograms window[7] where the various TIC/XIC chromatograms are displayed for the various mass spectrometry data files that have been loaded. There is, by definition, a single TIC chromatogram for each MS run data set currently loaded in the program. However, this window will also display ion current chromatograms that are computed as a result of an integration step from the other windows, like from the Mass spectra window or from the Drift spectra window. In this case, the chromatogram is an extracted ion current chromatogram (XIC chromatogram);
- The Mass spectra window, where the various mass spectra are displayed. A given mass spectrum may originate from a TIC chromatogram, from the MS run data set tableview window, from a drift spectrum, or even from a colormap;
- The Drift spectra window, where the various drift spectra are displayed. Like stated before, drift spectra can originate from a variety of places;

[7] TIC stands for "total ion current" and XIC stands for "extracted ion current".

- The TIC/XIC chrom. /Mass spec. colormaps window, that contains int = f(rt, *m/z*) colormaps, two-dimensional representations of the relation between the retention time and the mass spectral data. The intensities are represented as colors in a gradient map;
- The Drift spec. /Mass spec. colormaps window, that contains int = f(dt, *m/z*) colormaps, two-dimensional representations of the relation between the drift time and the mass spectral data (in ion mobility mass spectrometry experiments only). The intensities are represented as colors in a gradient map;
- The Console window, where the various messages or analysis data elements are displayed for the user to select, copy and paste in an electronic lab-book;

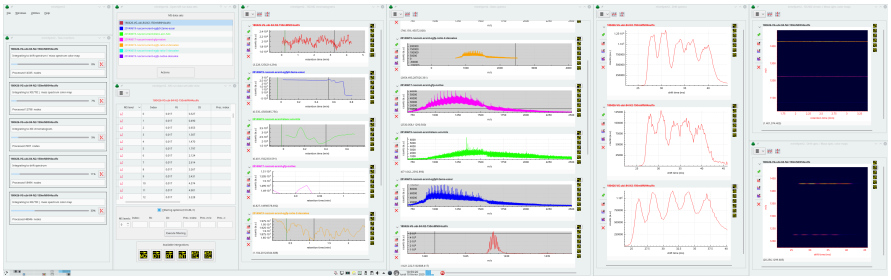


Figure 7: General View of the Graphical User Interface

The position and size of all the windows can be stored for a later session

4.3 The Main Program Window Menu

The menu bar in the main program window displays a number of menu items, reviewed below:

- **File**
  - **File⇨Open mass spectrum file(s) fully in memory**: Select the mass spectrum file(s) to load. The mass spectral data read from disk are stored in memory. This means that if the file is greater than the available system memory (RAM), the user is urged to select the next menu;
  - **File⇨Open mass spectrum file(s) in streamed mode**: Select the mass spectrum file(s) to load. The mass spectral data are read from disk, the TIC chromatogram and the MS run data set statistics are computed and then the data are freed. This file opening mode is compulsory when the mass data file is larger than the available system memory (RAM);
  - **File⇨Load mass spectrum from clipboard**: Create a mass spectrum from a textual representation of (*m/z*, i) pairs in the same format as described above for the txt,asc file format;

- **File⇨Analysis preferences**: Set the preferences for the exploration results storage;

- **Windows**

  The menus are self-explanatory, as they explicitely explain which window is to be shown. The **Save workspace** menu records on disk the position and size of all the windows, so that upon reopening the program, the windows all position themselves at the recorded position and size;

- **Utilities**

  - **Utilities⇨XIC chromatogram extractions**: Open a dialog window to set up XIC extractions for any one of the currently loaded mass spectral data set;

  - **Utilities⇨Isotopic cluster calculations**: Open the window in which the isotopic cluster calculations are configured and run;

  - **Utilities⇨Mass peak shape calculations**: Open the window in which the mass peak shape calculations can be configured and run;

  - **Utilities⇨Client-Server configuration**: Configure the communications between *MineXpert* and the companion software *MassXpert*, which may produce mass spectrometric data to be displayed by *MineXpert*.

- **Preferences**

  - **Preferences⇨Edit preferences**: Open the application preferences window (see Figure 2 – Application preferences window).

- **Help**

  This menu's items show help about the program itself and also about the Qt libraries that were used to build it. These informations are essential in case the user wants to make a bug report.

4.4 The Loaded MS Run Data Sets

Each time a new MS run data set is loaded from disk or from clipboard, a corresponding *MS run data set* item is added to the `Open MS run data sets` dialog window (see Figure 8 – The open MS run data sets).

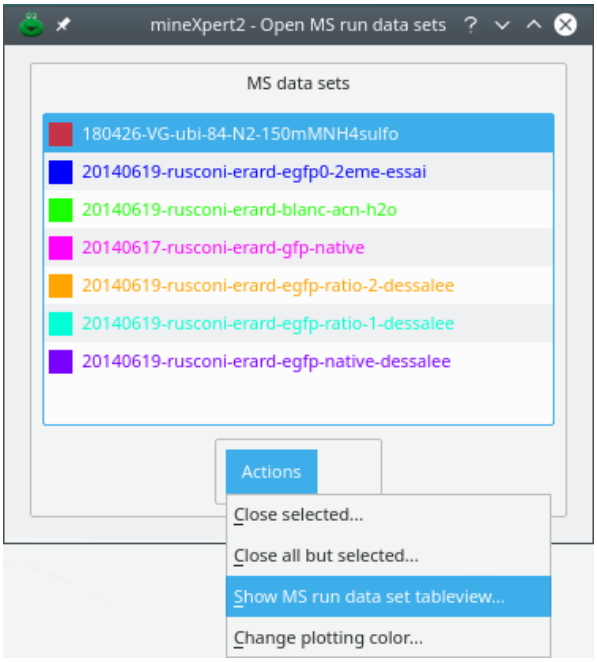


Figure 8: The open MS run data sets

Each time a new MS run data set is loaded, a corresponding item is added to the list widget. The drop-down menu lists a number of actions that can be performed with these items.

One of the most useful menu items available from the drop-down menu of Figure 8 – The open MS run data sets is the *Show MS run data set table view* item. That menu item triggers the display of a window containing a table view, like in a spreadsheet, that shows the data of each scan contained in the MS run data set loaded from file. This window is described in the following section (see Figure 9 – The table view-based scrutiny of one MS run data set).

4.5 Mass Spectral Data Listing in a Table View

From the Open MS run data sets dialog window, it is possible to select one list widget item corresponding to the data set of interest and select the *Show MS run data set table view* menu item that opens the MS run data set table view window shown in Figure 9 – The table view-based scrutiny of one MS run data set.

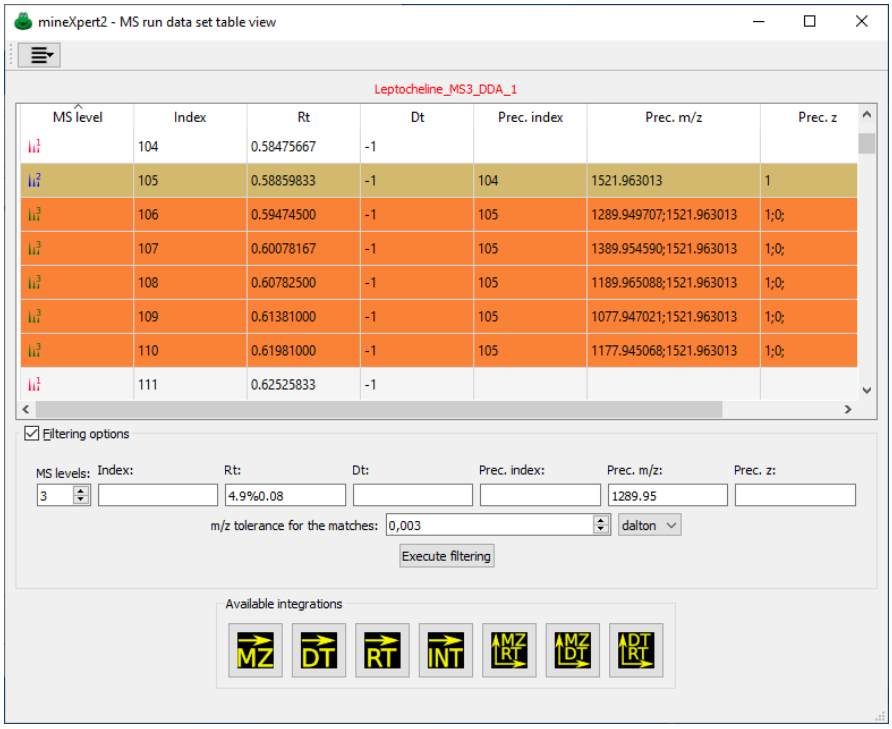


Figure 9: The table view-based scrutiny of one MS run data set

Each MS run data set might be scrutinized through the use of this table view. Each filtering criterion has a number of data insertion modalities, described in the text. In this example, the user has set a number of values in the Filtering options group box but has not yet executed the filtering, so the full mass spectral data set is listed in the table view.

The data that belong to each spectrum in the acquired mass spectral data set are shown as values in dedicated columns. Each row represents a mass spectrum of the data set. There might be tens of thousands spectra in an acquisition. Note that two columns may contain `;`-separated values. These columns are Prec. #mz; and Prec.z, respectively precursor ion *m/z* and precursor ion charge. Ions of differing *m/z* values and of differing charge values might be accumulated in a trap and then fragmented in a single fragmentation step. The mass data file thus contains spectra with a list of selected precursor ions.



Tip

When multiple values are listed in precursor *m/z* and charge cells, these values behave as if they were alone in the cell. When performing filtering of the data (see below), then the program treats each value of the list as if it were alone in the cell. For example, in Figure 10 – Mass spectral data set filtering using combined criteria the user filtered the MS data on the basis of a Prec. #mz; value of 1289.95. That filter selected the row that lists, as Prec. #mz;, the value 1289.952393;1521.973633.

In the MS run data set table view window, the checkable group box labelled Filtering options allows one to open a series of widgets in which to set the values for various filtering criteria.

- MS levels: Set 0 to retain all the mass spectra, irrespective of their MS level. Enter value 1 to filter out any MS level that is not 1. This fields has no practical limit, allowing one to perform MS^n data exploration.
- Index: Filter by the index number of the mass spectra. This may be of interest when the indices are somehow known in advance.
- Rt: Filter by retention time, that is, the time at which the mass spectrum was acquired, relative to the start of the mass data acquisition.
- Dt: Filter by drift time (ion mobility mass spectrometry generates this kind of data).
- "Prec. index": Precursor index.
- "Prec. *m/z*": Precursor charge.

The general syntax used to fill-in the filtering options is according to the following scheme:

- *simple value*, like 1 for MS levels.
- <value: keep rows where values are less than the entered value.
- >value: keep rows where values are greater than the entered value.
- >start <end: keep rows where values are in between start and end. The match is not inclusive.
- start--end: like before, but the match is inclusive. Mind the double-dash that delimits the values of the range.
- value%tolerance: equivalent to the preceding item, unless start is (value - tolerance) and end is (value + tolerance).

> **The specific case of precursor *m/z* values**
>
> When inserting a *m/z* value, there is a specific tolerance widget "*m/z* tolerance for the matches" that allows one to define the tolerance with which the matches for the precursor *m/z* values are performed. That mechanism does not override the value%tolerance value definition modality described above; instead, it will add another tolerance to the range values.

Once the filtering values have been entered and validated by clicking "Execute filtering" or by keying-in Ctrl Enter, the items (that is, the table rows) that verified the filter—still present in the table view—might be selected (all of them or only some of them) for their integration to a variety of destinations. The destination of each integration is selected

by clicking the proper icon push button from the button row below. The meaning of these push buttons will be described later (see Section 4.7.2 – The Right Column of Buttons Configures the New Integration to Run).



Tip

Make sure to set the *m/z* integration parameters before performing an integration to a mass spectrum. The setting dialog window opens up using the *Open* **m**/z *integration parameters dialog* menu item from the window's main menu, at the top left corner of the window. If the bin size value is too small, the integration might last very much longer than for a "reasonable" size.

The user might enter as many filtering criteria as necessary to pin point the most elusive feature in the mass spectral data set. In Figure 10 – Mass spectral data set filtering using combined criteria, the user has filtered the data to retain MS^3 mass spectra obtained by fragmenting ions of a given *m/z* value using a specified tolerance for the *m/z* match in Dalton units.

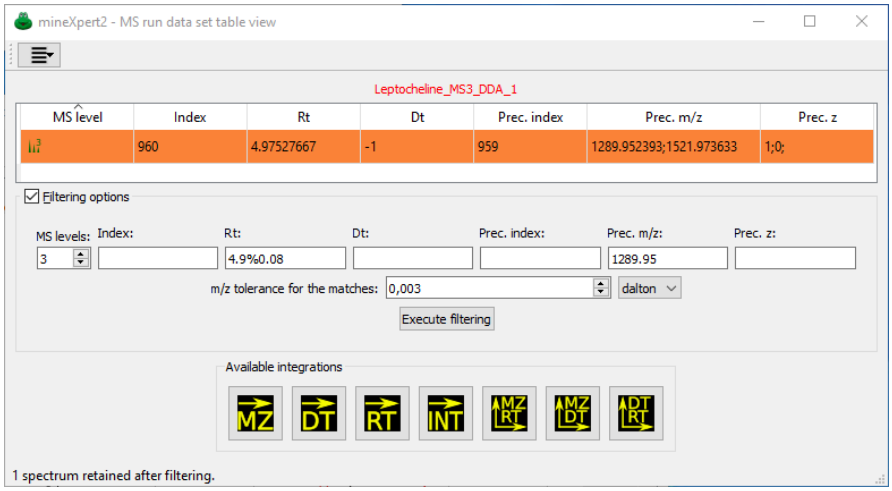


Figure 10: Mass spectral data set filtering using combined criteria

The use of combined filtering criteria allows one to pin point elusive mass spectral features.

4.6 The Main Data-Plotting Windows

This section will succinctly describe the main data windows of *MineXpert*. Each window will be described in greater detail when the features of the program will be described.

Tip

Although not visible, the various plot widgets in the various plot windows are glued together with splitters that allow their resizing if the window becomes too crowded, as shown in Figure 11 – Resizing the plot widgets with the splitter.

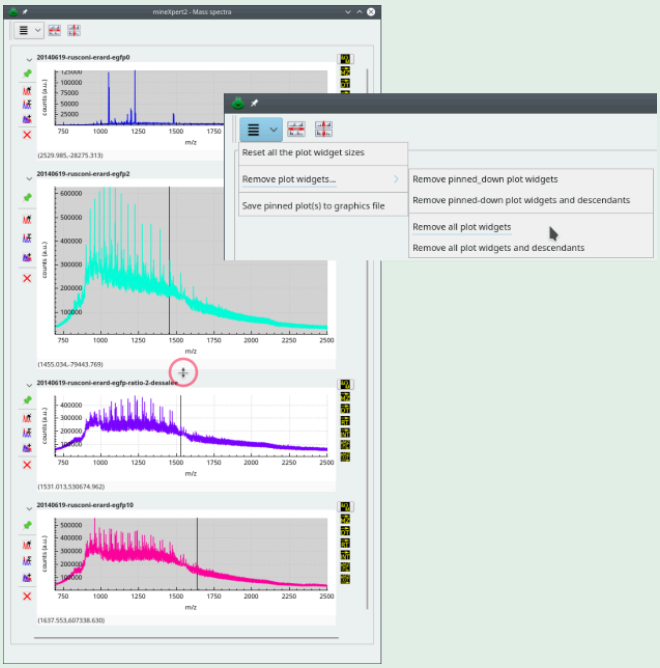


Figure 11: Resizing the plot widgets with the splitter

When positioning the mouse cursor between two plot widgets, the cursor switches to the splitter cursor shape (circled in red). Dragging that cursor will widen/shrink the plot widgets. To reset the plot widgets as they were initially created, use the *Reset all the plot widget sizes* from the main window menu shown on the right.

As visible in all the figures of the plot widget-containing windows described in the next sections, the general structure of these windows is to have a menu in the toolbar, icon buttons in that same toolbar and then plot widgets stacked below in the order in which they have been created.

Tip

In this document, the term “hamburger menu button” is used to describe the buttons that look like this: ☰

The window main menu (hamburger menu) offers menus that are all self-explanatory. The tool bar sports two buttons that:

- Lock the x-axis of all the plot widgets in the window (first button with the horizontal red double-head arrow across plots).
- Lock the y-axis of all the plot widgets in the window (second button with the vertical red double-head arrow across plots).

> **The multi-graph plot widget has gone**
>
> In the first version of *MineXpert* the plot window was divided into two parts; the upper part had a plot widget in which all the plots were overlaid, the lower part had as many plot widgets as there had been integrations because each plot widget could only have one plot in it.
>
> This is no more true now, because, as will be described later, it is now extremely easy to duplicate any plot into any plot widget so as to overlay any number of plots into any number of plot widgets.

4.6.1 The TIC Chromatogram Window

Each time a new mass spectrum file is loaded, its corresponding TIC chromatogram is computed and then displayed in a new plot widget in the TIC/XIC chromatogram window Figure 12 – The total ion current (TIC) chromatogram window. That window is indeed specialized in the plotting of TIC or XIC chromatograms. Each new TIC chromatogram plot that is generated as a result of the loading of a mass spectrometry data file is plotted using a new color. That color encodes the filiation of the whole set of plots that are generated while performing various integrations starting from that initial TIC chromatogram plot. For example, a red TIC chromatogram plot that serves as the starting point for a mass spectrum integration triggers the creation of a new mass spectrum trace plot that is plotted in red color. For colormap windows, where there is not trace plot, the axis and tick labels of the plot are of the same color as that of the initial TIC chromatogram plot.



Figure 12: The total ion current (TIC) chromatogram window

As many traces as necessary can be shown in the window. Each plot has its own set of crosshair markers.

Unconventional situations while loading mass spectrometry data files

- When the user loads mass spectrometric data from a non-profile (that is, non LC-MS) acquisition data file or from clipboard data, like when a mass spectrum is opened from a txt,asc,xy text-based format file where the data correspond to a *single* spectrum, *not* a sequence of spectra, the TIC chromatogram really has a single (rt,i) pair denoting the TIC intensity at the single retention time of that very unique spectrum. The TIC chromatogram window thus artificially creates and displays a TIC chromatogram that is a simple line, like shown in the bottom TIC chromatogram plot of Figure 12 – The total ion current (TIC) chromatogram window. From there, integration happens like in conventional situations. Make sure that the

mouse drag movement encompasses the whole TIC region by first unzooming a bit the trace.

- When loading data files that only contain MS^n data (with n > 1), like the proteomics-oriented *Mascot generic files* (MGF), *MineXpert* cannot compute a TIC chromatogram. Indeed, a TIC chromatogram can only be computed for spectra acquired in full scan mode, that is, MS data, not MS^n data (with n > 1). In this case, the TIC chromatogram plot that is created does not contain any plot. The only way the user can start exploring such kind of MS^n (n > 1) data is by using the MS run data set table view window (Section 4.5 – Mass Spectral Data Listing in a Table View).

4.6.2 The Mass Spectrum Window

The mass spectrum window is specialized with the plotting of mass spectra. It thus contains all the plot widgets that display all the mass spectra that were created as a result of mass spectral data integrations. These integration might have originated in any plot widget of any kind (mass spectrum, drift spectrum, colormap, for example). For example, the user might select a region in a TIC chromatogram and then ask that a mass spectrum integration be computed. In this case, the resulting mass spectrum is displayed in a new plot widget that is located in the mass spectrum window (Figure 13 – The mass spectrum window).

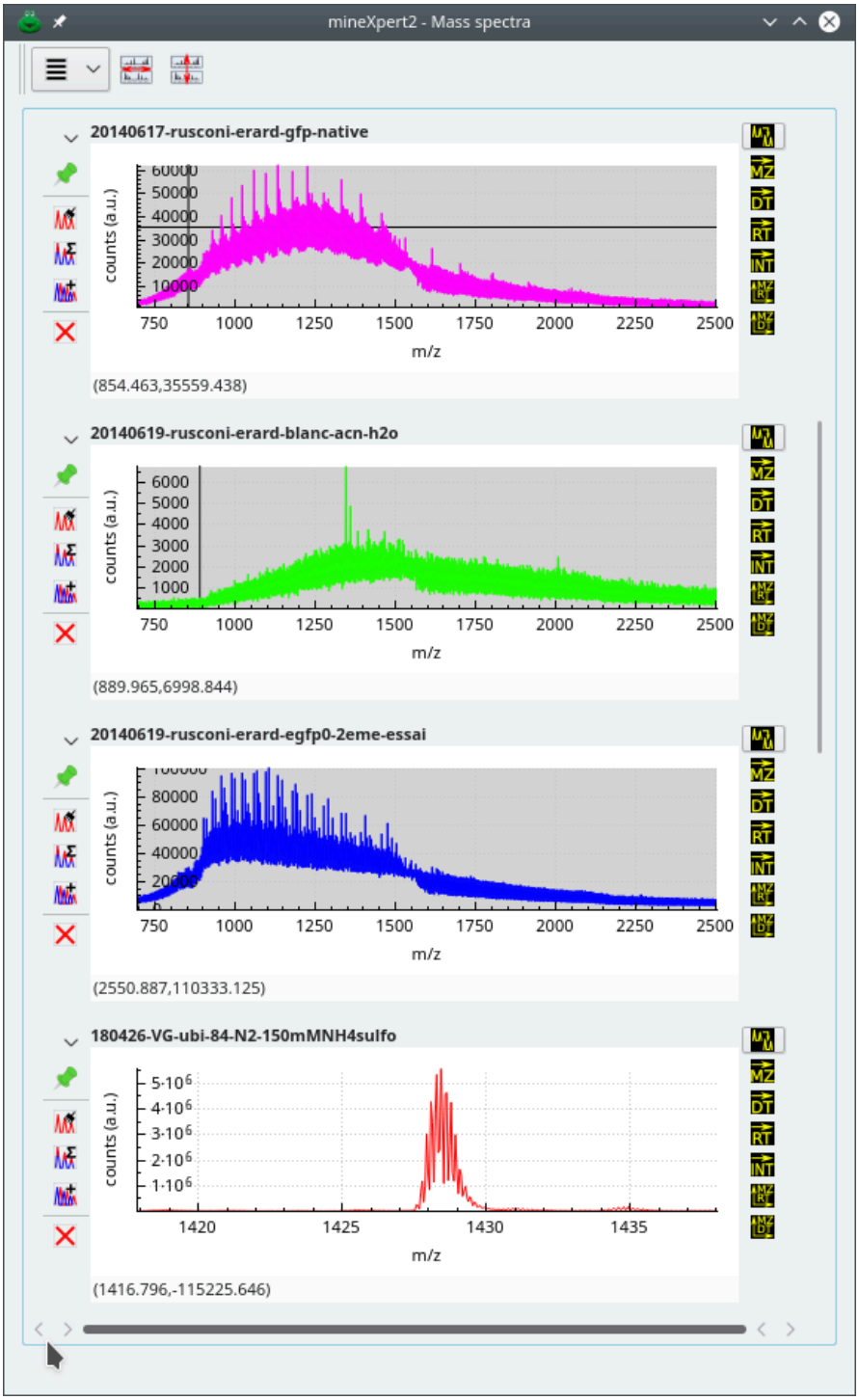


Figure 13: The mass spectrum window

This mass spectrum window shows multiple mass spectra.

4.6.3 The Drift Spectrum Window

As described for the mass spectrum window, the drift spectrum window contains all the plot widgets that display drift spectra (Figure 14 – The drift spectrum window).

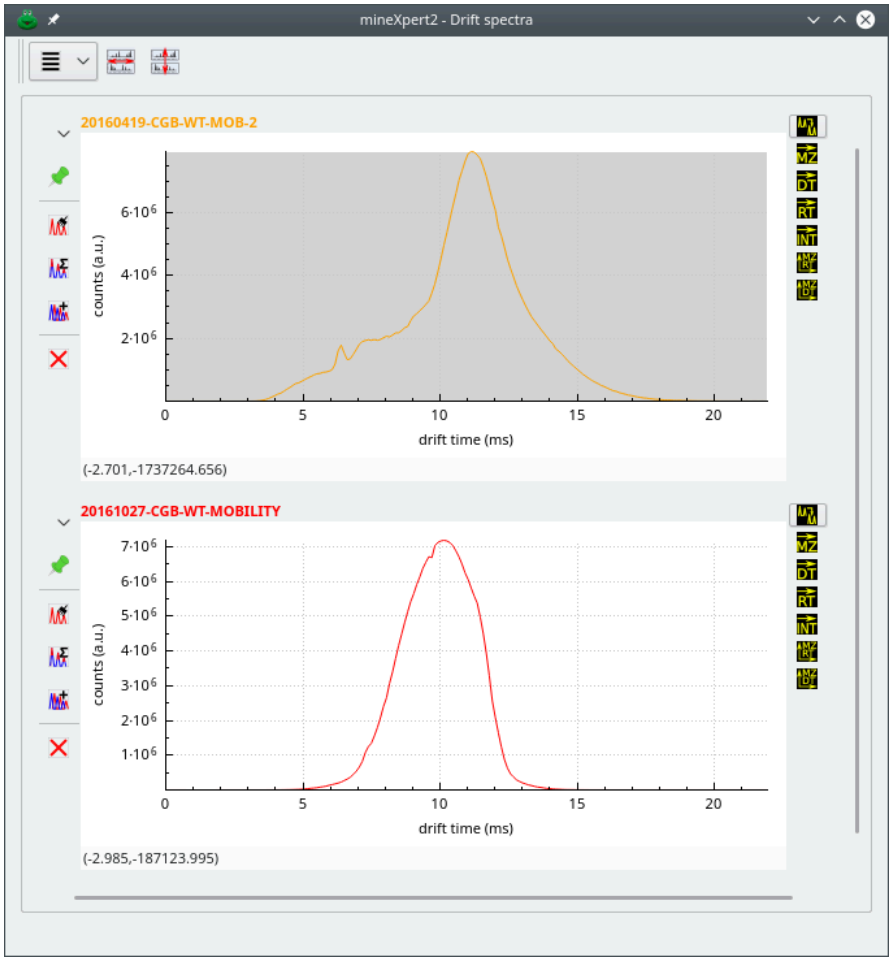


Figure 14: The drift spectrum window

This drift spectrum window shows multiple drift spectra.

4.6.4 The Retention Time *vs* Mass Spectrum Color Map Window

The int = f(rt, *m/z*) colormap window displays a heat map relating all the retention time values with the corresponding mass spectra (see Figure 15 – The int = f(rt, *m/z*) colormap window). The tool bar sports one more menu to transpose the axes of the all the color map plots contained in the widgets that are pinned-down.

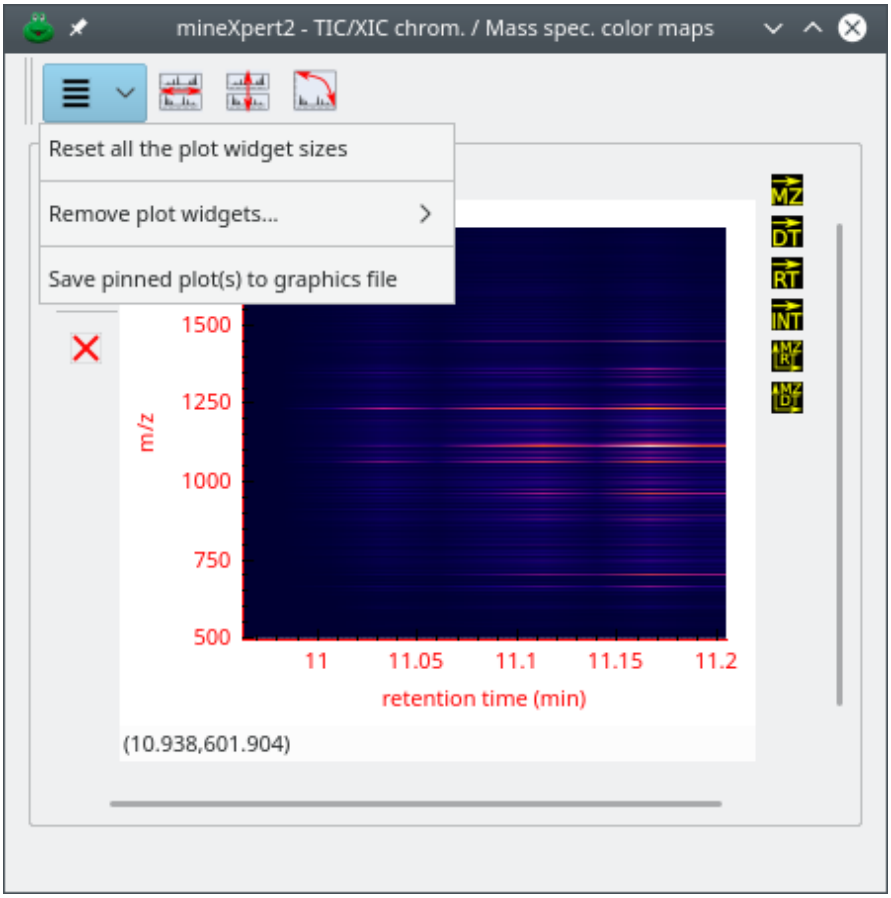


Figure 15: The int = f(rt, *m/z*) colormap window

The mass spectrum *vs* retention time colormap window relates mass spectra (y-axis) to retention times (x-axis).

4.6.5 The Drift Time *vs* Mass Spectrum Color Map Window

The int = f(dt, *m/z*) colormap window displays a heat map relating all the drift time values with the corresponding mass spectra (see Figure 16 – The int = f(dt, *m/z*) colormap window).

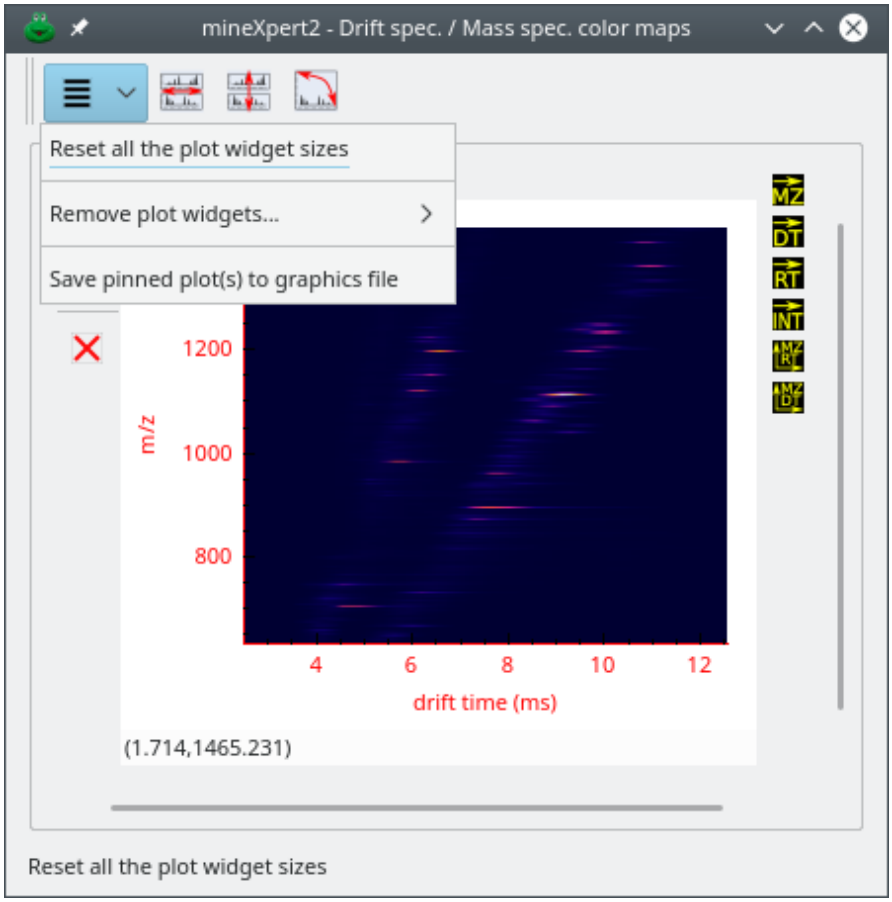


Figure 16: The int = f(dt, *m/z*) colormap window

The mass spectrum *vs* drift time colormap window relates mass spectra (y-axis) to drift times (x-axis).

4.6.6 The Drift Time *vs* Retention Time Color Map Window

The int = f(rt, dt) colormap window displays a heat map relating all the drift time values with the corresponding retention time values (see Figure 17 – The int = f(rt, dt) colormap window).

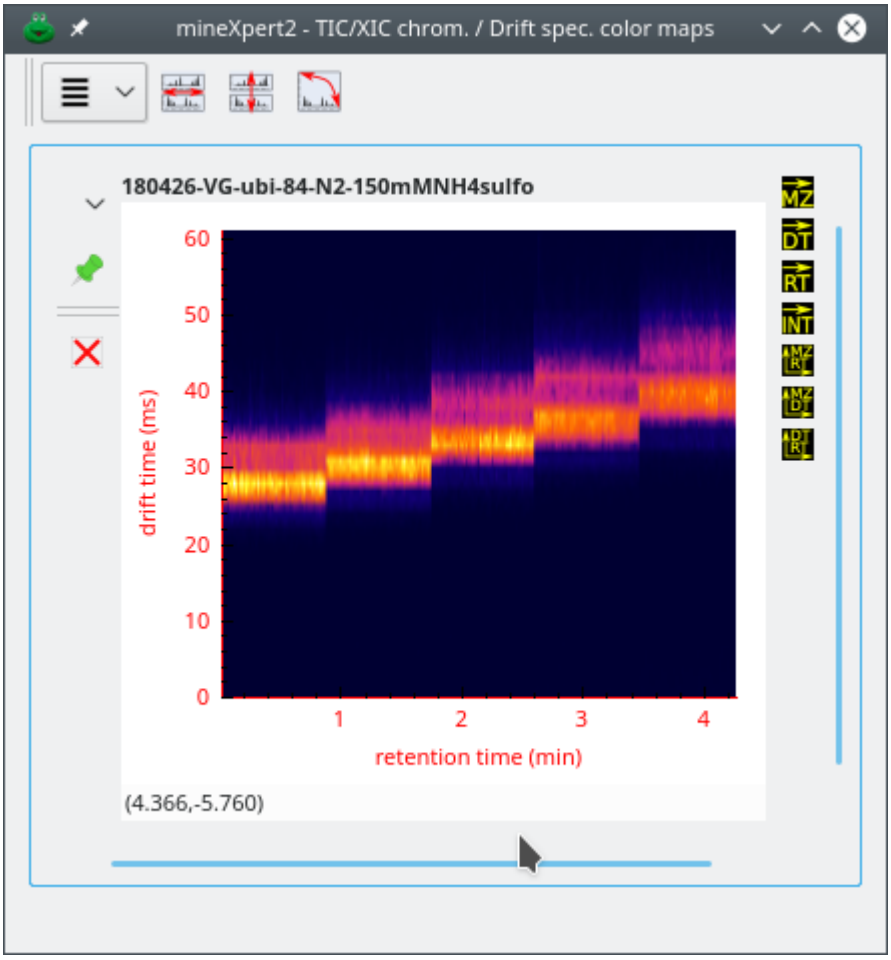


Figure 17: The int = f(rt, dt) colormap window

The drift spectrum *vs* TIC chromatogram colormap window relates retention times (y-axis) to drift times (x-axis).

4.6.7 Exploring data in multiple plot widgets

When comparing plots (be them trace plots or colormap plots), it is often necessary to align the plots according to the x-axis so as to align features of interest in the various vertically-stacked plot widgets. In such a situation, there is now way to move the cursor through all the plots simultaneously so as to read the x-axis value while marking that position with the crosshair cursor. To enable a vertical mark over all the aligned features in all the plot widgets, click the Figure 18 – Activation of the vertical rule icon.



Figure 18: Activation of the vertical rule

When this toolbar' third button is checked (the icon shows a dotted vertical red line across multiple widgets), a horizontal slider appears at the top of the window. When that slider is moved, a vertical rule is drawn at its position. The rule is drawn across all the plot widgets in the window.

If the x-axis range is locked (when first toolbar button in Figure 18 – Activation of the vertical rule with two-arrow-headed red horizontal line is checked) and the vertical rule button is checked also, the vertical slide rule continuously updates the x-axis value near the slider handle. See Figure 19 – Using the vertical rule across all the plot widgets for an illustration.

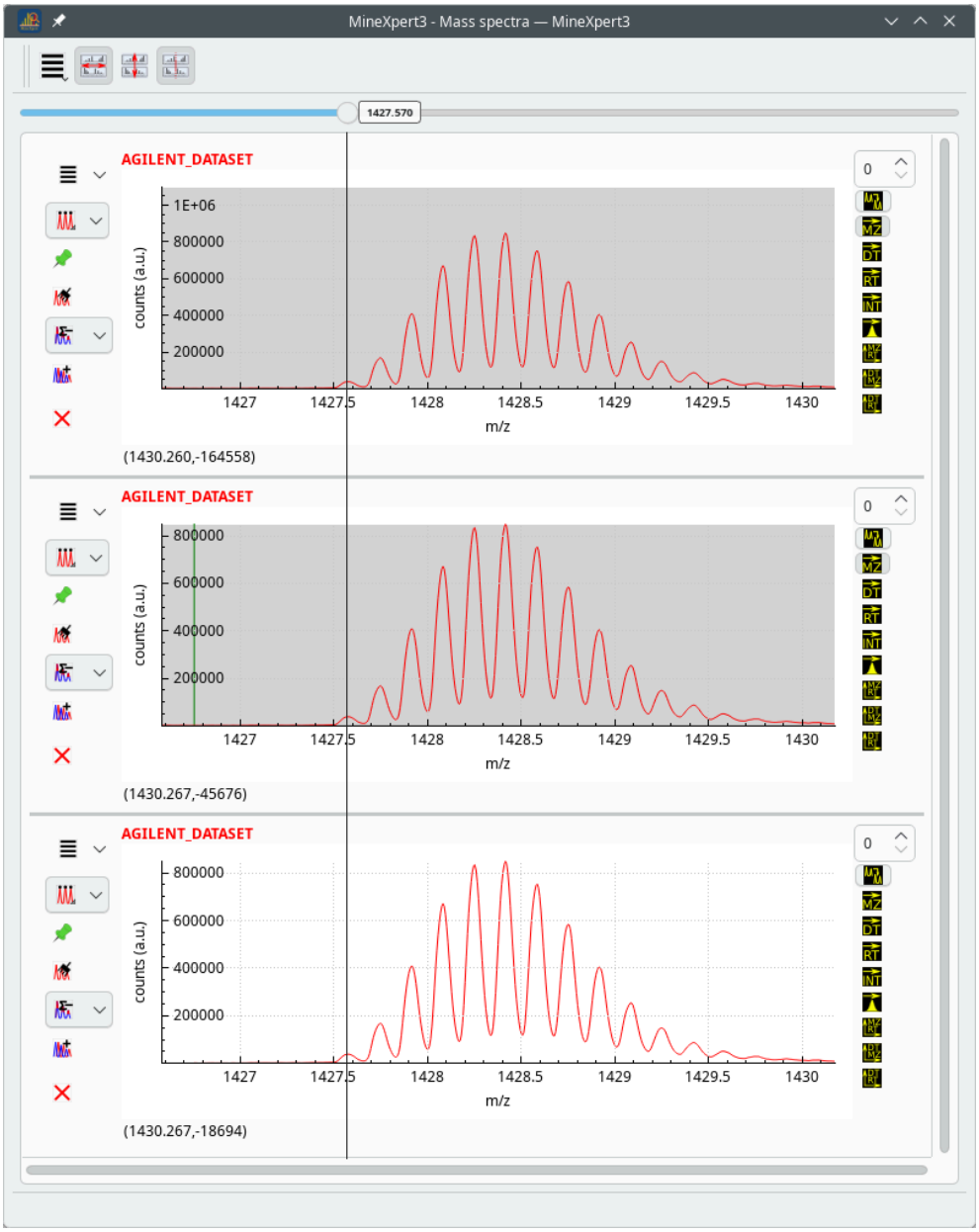


Figure 19: Using the vertical rule across all the plot widgets

If the x-axis range is locked while having the vertical rule slider moved, a label is shown at the slider handle with the x-axis value at the vertical rule position.

4.7 General Structure of the Plot Widgets

The TIC/XIC chromatograms, mass spectra, drift spectra, retention time *vs* mass spectrum and drift time *vs* mass spectrum colormap windows (collectively called *plot widget windows*) are all structured in a similar way as described below.

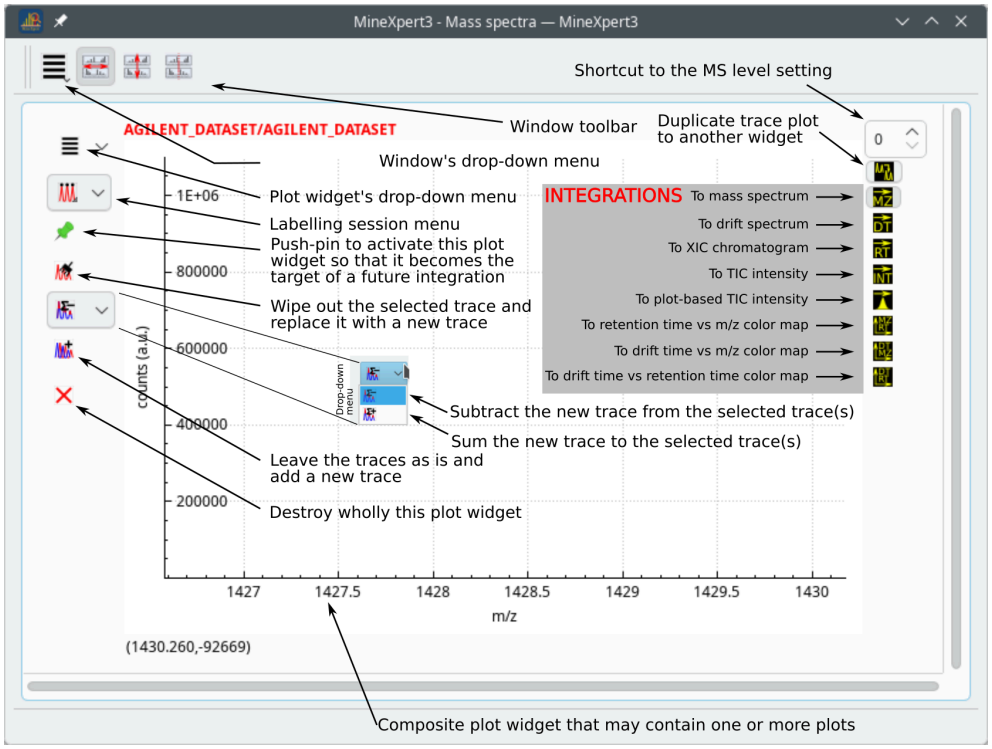


Figure 20: Description of the various widgets that make a composite plot widget

A composite plot widget is a widget that contains a number of other widgets. These "sub-widgets" are described.

The window in Figure 20 – Description of the various widgets that make a composite plot widget contains a single composite plot widget delimited in the figure by a light blue rectangle. Each composite plot widget's plotting area is sided by two columns of icons, which really are, for most of them, check buttons. The buttons in the left column configure if and how this plot widget is going to be the *receiving container* of a new integration result (result that takes the form of a graph plot, in contrast to a colormap plot). The buttons in the right column configure *what kind of integration* is to be performed using, as a starting point, one of the plots of this plot widget. Indeed, as already mentioned above, integrations in any plot widget might be configured to generate any kind of data: a XIC chromatogram, a mass or drift spectrum... If the data generated by a given integration constitute a XIC chromatogram, that chromatogram will be plotted in the TIC/XIC chromatograms window.

Both columns of buttons (left and right) are detailed in the next section because these constitute the basis of the operation of *MineXpert*.

4.7.1 The Left Column of Buttons Configures the Reception of New Data

For this description, we will assume that the user is working in the TIX/XIC chromatograms, window and is performing a new integration to a mass spectrum. The result of that new integration is thus data that need to be installed in the Mass spectra window, as depicted

in Figure 20 – Description of the various widgets that make a composite plot widget. Such a window, however, might already contain a number of plot widgets; it is thus necessary to establish where the new plot is going to be created and how it will be created. These two questions are answered in detail below by describing the widgets in the left column of widgets in a composite plot widget.

- The composite plot widget's drop down menu (button with the downward arrow) will be described in detail later. The menu options are actions that apply to the plot widget that owns the menu;

- The labelling session menu button ( ) opens a menu that provides features about labelling peaks in the graphs plotted in the composite plot widget;

- The push-pin button, if it is pushed (therefore colored in red), indicates that the plot widget "claims" to be the receiving container for any new integration data destined to the window containing it.

> **The expression "*pinned-down widget*"**
>
> A plot widget having its push-pin pushed down is said to be *pinned-down*. This is a terminology that will be used extensively in the rest of this manual.

  If the window contains more than one plot widget *and* if more than one plot widget are pinned-down ( ), then all these plot widgets claim to be the receiving container of any integration data. If no plot widget is pinned-down, the data will be plotted in a new composite plot widget that will appear at the bottom of the plot widget window, that is, below all the other composite plot widgets.

  In the list items below, we assume that the composite plot widget containing the various check buttons is pinned-down and belongs to a non-color-map plot window.[8]

- Checking the check button indicates that the new integration data will be plotted in the pinned-down composite plot widget an will *replace* any selected plot in the plot widget.

- Checking the or the check button indicates that the new integration data will be *subtract-combined* or *add-combined* from or into any selected plot in the plot widget. This feature is particularly useful when trying a step-wise estimation of the mass spectral data out of discrete discontinuous features like TIC chromatogram peaks or drift spectrum peaks. Likewise and reciprocically, it might be useful to extract, from different

[8] Combination of colormaps is not yet implemented, although they might be useful in some situations.

mass spectral features in a given mass spectrum, the combined TIC chromatogram for them. Typically, when an analyte appears in a mass spectrum under different charge states that are separated by non-baseline regions, being able to assess the XIC chromatogram for them in a series of combined integrations is extremely useful. The subtract-combination is typically used for removing background from a given trace using other blank trace as the background level.

- Checking the  check button indicates that the new integration data will be plotted in a new plot that will be added to the composite plot widget.

4.7.2 The Right Column of Buttons Configures the New Integration to Run

Each composite plot widget may contain zero or more plots (also called graphs or bars, depending on the kind of data). As mentioned earlier, any plot can serve as the starting point for a new integration to the same or another kind of data. For example, from a plot in the TIC/XIC chromatograms window, one might want to integrate a given retention time range to a mass spectrum or to a drift spectrum. How are those destinations determined? The destination of an integration is selected by pressing one of the buttons in the right column of check buttons described below.

- The top widget of the right hand side column of widgets is a spin box widget where the user can set the MS level of the next integration to be performed. Setting a value in this widget is equivalent to setting the same value in the MS level field of the MS fragmentation specification setting dialog window (Figure 37 – Setting the MS^n fragmentation parameters at Section 5.1.2 – Setting the MS^n Fragmentation Parameters).

-  This button is an exception, because it actually is *not* a check button. It is a true button that, when clicked, elicits the duplication of the currently selected trace(s). There is no integration going on here: the plot(s) is(are) copied "as is" with no processing whatsoever into a new composite plot widget that is created beneath all the existing ones. If more than one graph is currently selected, each graph is duplicated into its own composite widget.

-  This is the check button that, when pressed, directs the new integration to produce mass spectral data to be directed to the Mass spectra window. Depending on the configuration of the new integration data fate, the new data may either be plotted to a new composite plot widget or be plotted as a new plot in any pinned-down composite widget or be added/subtracted from any plot in any pinned-down plot widget (see Section 4.7.1 – The Left Column of Buttons Configures the Reception of New Data). Inte-

gration to a mass spectrum implies configuring how the *m/z* values are to be combined (with or without binning and, if binning is required, the size of the bins). Please, refer to Section 5.1.3 – Setting the *m/z* Integration Parameters for a detailed explanation.

- DT This is the check button that, when pressed, directs the new integration to produce ion mobility spectral data to be directed to the Drift spectra window. "DT" stands for "drift time".

- RT This is the check button that, when pressed, directs the new integration to produce XIC chromatogram data to be directed to the TIX/XIC chromatograms window. "RT" stands for "retention time".

- INT This is the check button that, when pressed, directs the new integration to produce a *single TIC intensity value* that is displayed at the bottom of the plot widget, in the widget's status bar.

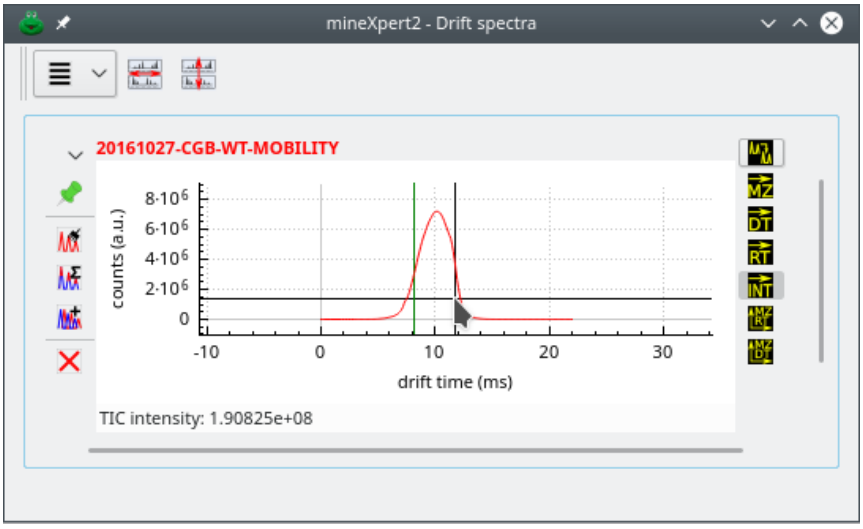


Figure 21: Single TIC intensity value for any mass spectral data feature

When using the *single TIC intensity value* function, any feature of a plot will be integrated to a single TIC value that represents the sum of all the TIC values for all the MS data in the MS run data set that match the feature selection. The obtained single TIC intensity value is displayed at the bottom of the plot widget in which the calculation was requested. In this example, the feature that was selected for integration is located between the vertical markers.

- This is the check button that, when pressed, directs the new integration to produce a *single TIC intensity value* that is displayed at the bottom of the plot widget, in the widget's status bar, as described above. The distinction is related to the fact that integration to an intensity value is performed on the basis of the curve data points and not by looking into the initial mass spectral data set loaded from disk.

- MZ RT This is the check button that, when pressed, directs the new integration to produce int = f(*m/z*, rt) data to be directed to a colormap composite plot widget. This function

is only useful for mass spectrometric data sets that were acquired in "profile" mode; that is, for acquisitions performed along a time frame, like an on-line chromatography-MS experiment.

- This is the check button that, when pressed, directs the new integration to produce int = f(*m/z*, dt) data to be directed to a colormap composite plot widget. This function is only useful when the mass spectra were acquired during an ion mobility mass spectrometry experiment.

- This is the check button that, when pressed, directs the new integration to produce int = f(dt, rt) data to be directed to a colormap composite plot widget. This function is only useful for mass spectrometric data sets that were acquired in an ion mobility mass spectrometry experiment.

4.7.3 The Composite Plot Widget Main Menu

There are two kinds of composite plot widgets:

- The composite plot widgets that host either or both graph (trace) or bars plots. The widgets host plots that represent mass spectra (graph plots), drift spectra (that is, ion mobility spectra, also graph plots) or centroided mass spectra (bars plots);

- The composite plot widgets that host only colormap plots. The colormap plots are used to plot complex data where the intensity is a factor of two variables, like int = f(dt, *m/z*), where the two-dimensional colormap plot represents color-based intensities as a function of both ion mobility spectral data in one dimension and mass spectral data in the othe dimension.

In the two cases above, the composite plot widget menu is not the same because the features of the plots are not the same.

4.7.3.1 The Graph|Bars Composite Plot Widget Main Menu

The graph or bars plot widgets can host any number of plots, be them ion mobility or mass spectra, TIC|XIC chromatograms or centroided mass spectra. A graph (or trace) plot is a curve-type plot, that is, a two-dimensional where the data points are connected to each others by a line which draws the curve. A bars plot is a plot where each data point stands on its own in the plot as a bar. The typical bars plot is a centroided mass spectrum plot. In any of these two plot types, the ordinate value is an intensity value for the given abscissae value (be it a *m/z*, a dt, or a rt value).

The main composite plot widget's menu is described in Figure 22 – Trace plot widget main menu.

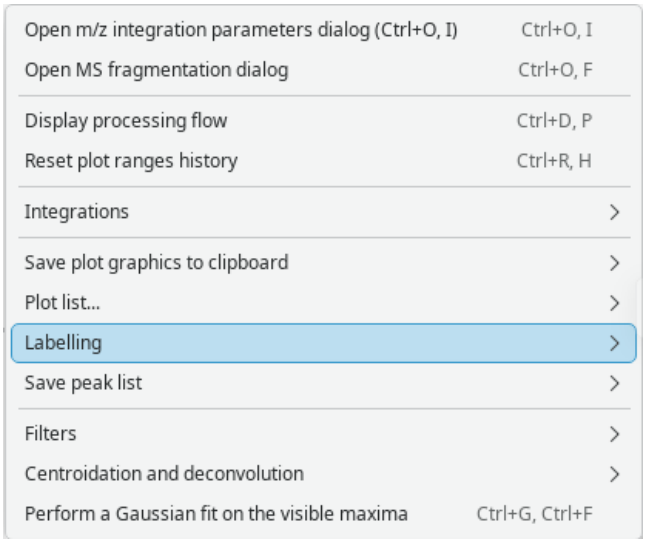


Figure 22: Trace plot widget main menu

The composite plot widget's main menu provides actions that may apply only to the plot widget that owns the menu.



Tip

The menu items described below perform actions that apply to individual plots inside the plot widget that owns the menu. *MineXpert* tries to understand the will of the user: when only one plot is in the widget, the menu action applies to that plot, even if it is not currently selected. As soon as there are more than one plot in the widget, the action requires that the target plot be selected. If no plot is selected, a dialog window will instruct the user to select the specific plot(s) of interest.

- *Open m/z integration params dialog*: Open a dialog to set the *m/z* integration parameters (see Section 5.1.3 – Setting the *m/z* Integration Parameters).
- *Open MS fragmentation params dialog*: Open a dialog to set the MS^n requirements for the MS data integration (see Section 5.1.2 – Setting the MS^n Fragmentation Parameters).
- *Display processing flow*: Open a dialog and display the processing flow of the currently selected graph. If there is only one graph in the plot widget, the processing flow for that graph (even if it is not selected) is shown.
- *Reset plot ranges history*: Delete all the plot navigation milestones from the history and start anew with the current plot display. It won't be possible to iterate up in the plot ranges history using the `Backspace` key.

- *Integrations*: Configure the integrations to sum up the intensities of successive integrations (Section 5.8 – Cumulative Intensity Integrations) or to be performed in point-by-point mode (click on the graph plot at the place where point-by-point integration needs to start, then use the F3 to integrate the next plot point).



Figure 23: Submenu to configure the integrations plot widget main menu

The Integrations submenu defines the way integrations are to be performed.

- *Save plot graphics to clipboard*: Save the whole plot widget as a graphics object to the clipboard. The graphics file formats available are raster or *SVG*.

- *Plot list...*: Lists all the plots in this composite plot widget:

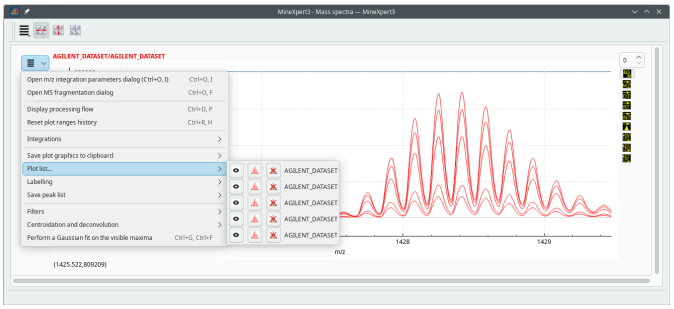


Figure 24: Submenu to list the plots in the composite plot widget

The three buttons attached to each plot name allow one to show/hide, select/deselect or destroy the corresponding plot.

- *Labelling*: Provides features to handle label lists associated to this composite plot widget's plots:

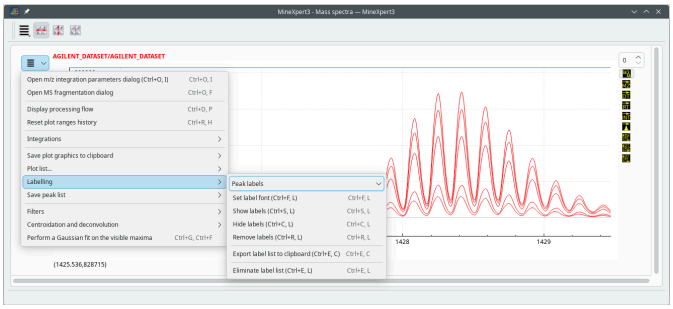


Figure 25: Submenu to manage the lists of labels

The different menus apply to the label list currently selected in the drop down list at the top of the submenu.

- *Save peak list*: Save plot data to the clipboard according to the modality shown:

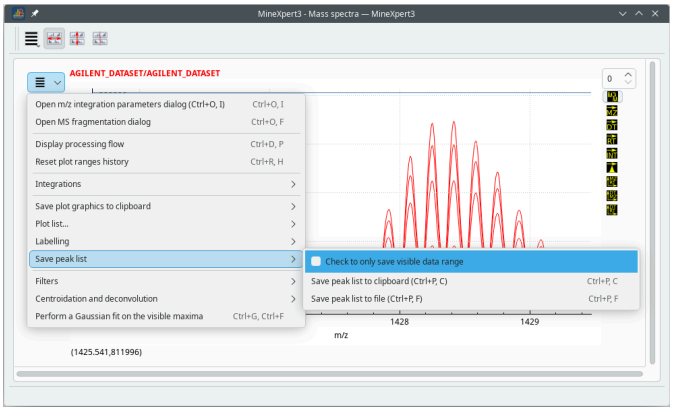


Figure 26: Submenu to manage the export of the plot data points

The export to file or clipboard can be restricted to the visible abscissae range of data in the current view of the plot widget.

- *Filters*: Apply filters to the plot.

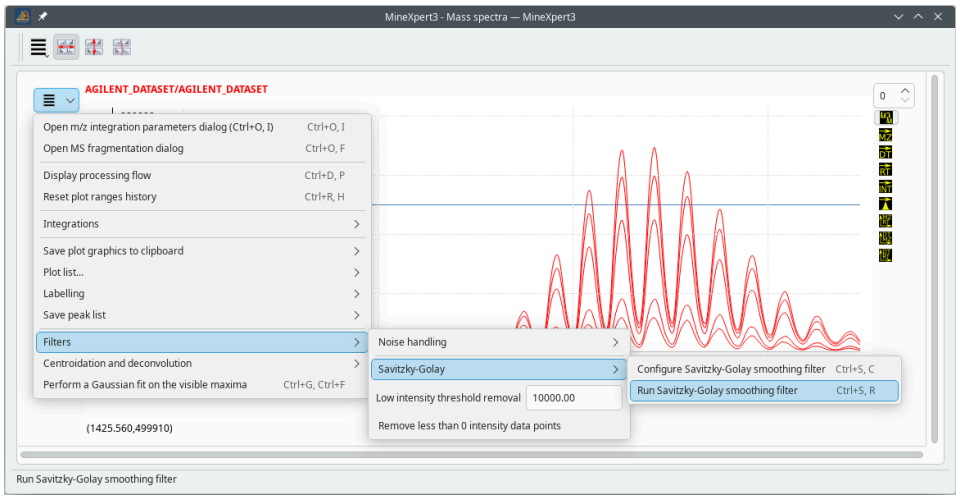


Figure 27: Submenu to select filters

The Savitzky-Golay filter requires previous configuration.

The Savitzky-Golay smoothing filter will be run with the configuration settings configured as at Section 5.1.6 – Savitzky-Golay Filtering of Any Kind of Data. The smoothed trace will be added into any pinned-down plot widget of the receiving window. If no plot widget is pinned-down, then a new plot widget will be created.

> **Tip**
>
> The composite plot widget's main menu will have menu items that differ depending on the kind of data being plotted in the widget. For example, it makes no sense to perform centroidation on a TIC/XIC chromatogram or on an ion mobility spectrum.

- *Centroidation and deconvolution*: This menu is specific of the `Mass spectra` window where all the composite plot widgets host mass spectral data. It is the subject of an in-depth review at Section 6 – Centroid Extraction from Mass Spectra and at Section 7 – Mass Spectral Deconvolutions.

- *Perform a Gaussian fit on the visible maxima*: The Gaussian fit will be performed on the selected plot (if there are multiple plots).

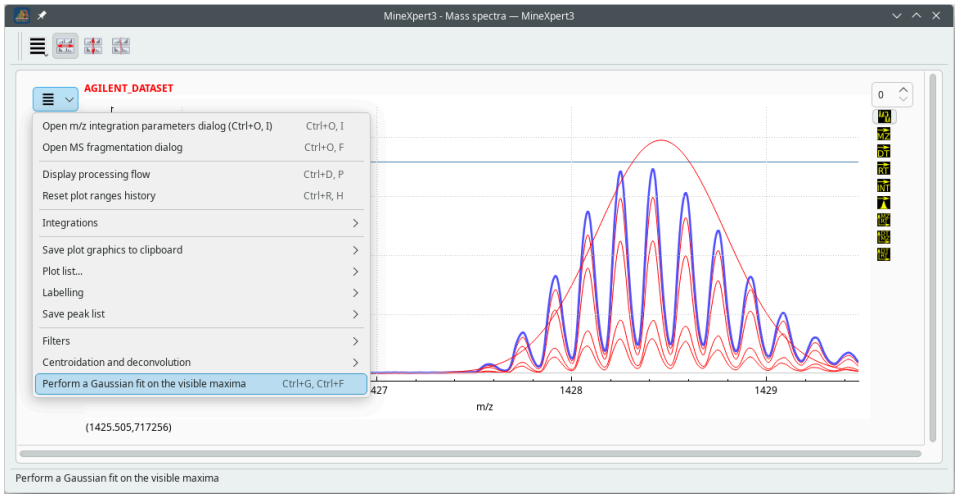


Figure 28: Fitting a Gaussian curve on an isotopic cluster

If there are multiple plots, the Gaussian fit will be performed on the selected one.

4.7.3.2 The Color Map Composite Plot Widget Main Menu

The processes described in this section are also described at Section 5.4 – Signal Enhancement and Noise Reduction in Color Map Plots.

The colormap composite plot widgets can host only one colormap plot since it is not possible to overlay another colormap onto an existing colormap. A colormap is a raster-type graph, that is, a bitmap image of which the third dimension is the intensity of the pixel at any given position in the image. That intensity is represented as a color. The colormap plots all share the same main plot widget menu that is significantly different than the menu available in graph|bars composite plot widgets.

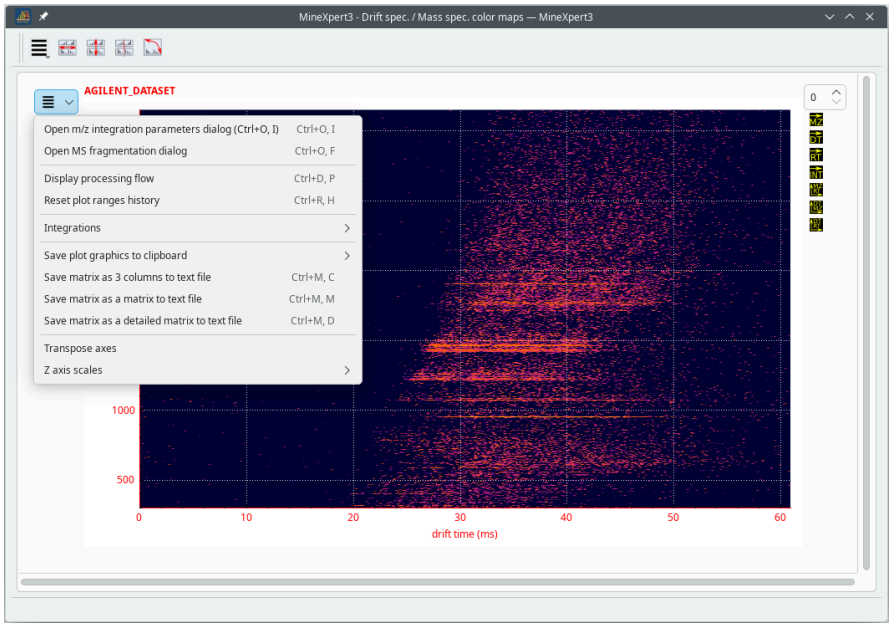


Figure 29: Colormap composite plot widget main menu

This menu contains specific menu items that are only of use on color map plots. The menus about data export are self-explaining.

- *Transpose axes*: Transform the data matrix to switch axes: the abscissae axis becomes the ordinate axis and vice versa.
- *Z axis scales*: Menu to handle the data representation of the colormap plot by modifying the intensity scale (the z axis scale, which will effect the colors of the colormap):

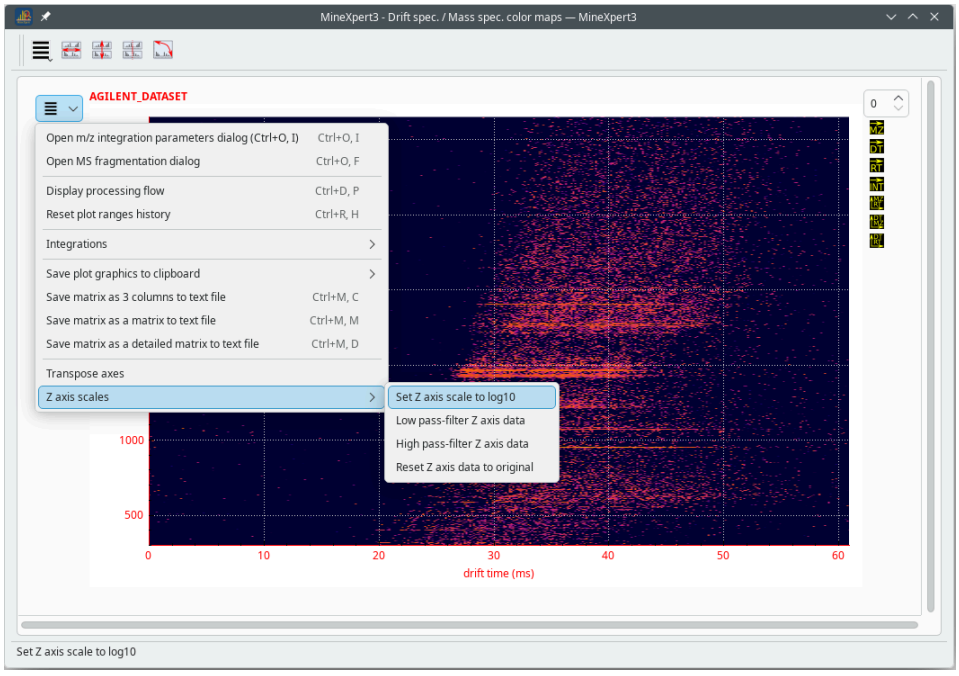


Figure 30: Submenus available to modify the z axis scale of intensities

The submenus are self explanatory.

When the *Set Z axis scale to log10* menu is triggered on a colormap plot that has not a highly dynamic intensity range (top plot), the signal becomes perfectly visible (bottom plot) in Figure 31 – Colormap plot widget main menu.

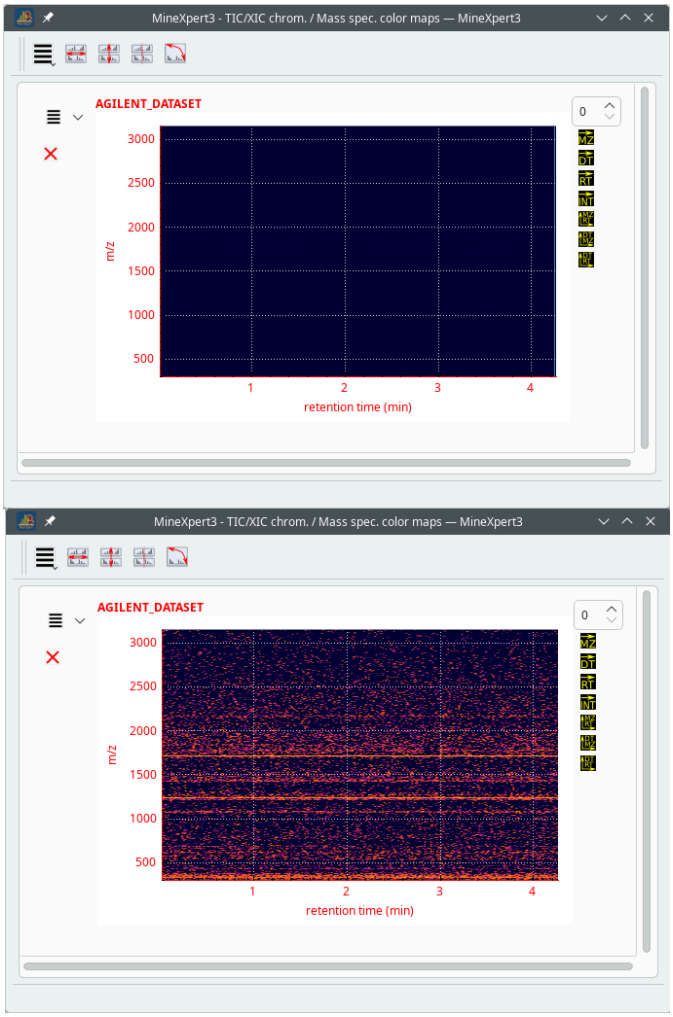


Figure 31: Colormap plot widget main menu

The switch to a log10 scale for the intensities might reveal a lot of missing signal when the intensities are used in a linear scale. The drawback is that noise can be hugely enhanced. See below for a way to reduce that noise.

- *Low-pass filter Z axis data*: The value (typically 90 %)that is entered in the input box that is shown is a *percentage* of the maximum intensity over the whole map. The function first computes a threshold using the formula

  threshold = percentage * max_int / 100.

  All the data points having an intensity value below that threshold are unchanged (they pass), while all the data points having an intensity value above that threshold have their value set to that threshold value (sort of capping the highest intensity values). This function is used to reduce the higher intensity dynamic range of the map and potentially enhance features of interest.

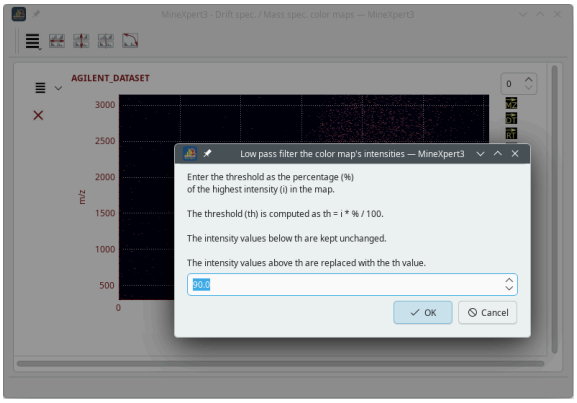


Figure 32: Low-pass intensity filter configuration for colormap plots

The dialog window explains the calculation and the outcome.

- *High-pass filter Z axis data*: The value (typically 20 %) that is entered in the input box that is shown is a *percentage* of the maximum intensity over the whole map. The function first computes a threshold using the formula

  threshold = percentage * max_int / 100.

  All the data points having an intensity value above that threshold are kept (they pass) while all the data points that have an intensity value below the threshold have their intensity set to that threshold value. This filter might be used to remove noise.

4.8 General Operation of the Plot Widgets

All the plot widget-containing windows, like the TIC/XIC chromatograms, Mass spectra and Drift spectra windows, all contain plot widgets that have a general working scheme for the way the plotted data can be visualized and the various plots are managed.

4.8.1 General Data Exploration Methods

The main visualization operations are described below. The following convention will be used to describe the mouse buttons:

- : left mouse button;
- : middle mouse button;
- : right mouse button.

> **Note**
>
> All the visualization operations are performed using the *left* mouse button (🖱). Keyboard keys are used to better qualify the zoom-in/-out or panning operations.

The different plot or colormap graph visualization methods are detailed below:

- Zooming-in and zooming-out:
  - Zoom-in: 🖱-click-drag to draw a selection rectangle. When the mouse button is released, the new plot view contains the data contained in the selection rectangle. Note that if the mouse drag over the y-axis spans less than 10 % of the vertical scale, the rectangle does not show up because the software thinks that the mouse drag operation is only horizontal (which has a meaning described later).
  - Zoom-in: `Ctrl` +🖱-click-drag along the x-axis over the region to zoom and release the mouse button. The new zoomed view contains the mouse drag-spanned region. The y-axis spans the region contained between the initial plot bottom y-value and the position on the y-axis of the mouse during the drag operation.

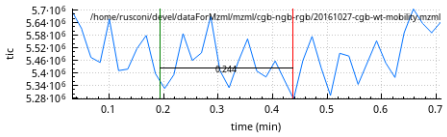


Figure 33: Zooming-in operation

The start and end x locations are visualized with green and red markers, respectively. The x-axis distance between the markers is refreshed all along the mouse movement.

  - Zoom-in/-out: `Ctrl` +🖱-click-drag *onto* either the x- or the y-axis to interactively zoom-in or -out along that selected axis. In this mode, the zoom operates by contracting/expanding the data in such a manner that the left/bottom part of the graph (the origin of the graph) is anchored and does not move. When the drag occurs towards larger values on the clicked axis, the view is zoomed-in along that axis. Conversely, it is possible to zoom-out by dragging the mouse towards lower axis values.
  - Zoom-in/-out: The 🖱-wheel-rotation can be used to zoom-in or -out the whole plot on both the x- and y-axis simultaneously. Note that the position of the mouse cursor when the wheel is rolled defines the new view of the plot. Practising a bit allows to make that zooming-in/-out mode very powerful.

- Zoom-out: To reset the zoom along one axis, 🖱-double-click that axis. In this case, only the clicked axis will be full-scale, the other axis remains unchanged. To reset the zoom on both axes in one go, 🖱-double-click one of the axes maintaining the `Ctrl` key pressed;

- Panning: 🖱-click-drag on one of the axes to pan the plot view along that axis;

- Relative full-scaling along the y-axis: in almost all of the zooming operations described above, it is possible to ensure that the y-axis is full-scaled automatically to the most intense peak visible in the zoomed region by pressing the `Shift` key.

- Recording the history of zoomed views:

  Each time a new zoomed-in/-out view of the plot is triggered, a history element is stored in the plot widget. To back-replay the various steps of the zoom-in/-out operations in sequence, from pre-last to first, hit the `Backspace` key. The exceptions to this mechanics is when the plot view is zoomed using the mouse wheel.

> **Tip**
>
> By pressing the `Shift` keyboard key along with the various 🖱-click-drag operations, the zoomed-in region will be displayed systematically in full y-axis range for the most intense data point in the new zoomed-in x-axis range.

4.8.2 Locking All the Composite Plot Widgets Views on X and/or Y Axes

The tool bar located at the top of all the plot widget-containing windows contains buttons to configure the locking of the X and/or Y axes so that all the plot widgets are synchronizedo over the axis of interest:

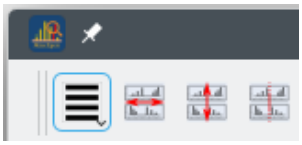


Figure 34: Plot window tool bar

The first two tool bar buttons configure the locking of the X axis and of the Y axis. The third button shows a vertical rule that allows the operator to check for precise alignment of features in all the plot widgets displayed in the plot window.

4.8.3 Checking for Precise Alignment of Features Across Composite Plot Widgets

It very often happens that one wants to ensure a given feature is located precisely at the same X-axis position in a number of plots of a plot window. To ease this process, one triggers the third button in Figure 34 – Plot window tool bar, which shows the rule visible at Figure 35 – The vertical rule marker spans the whole window.

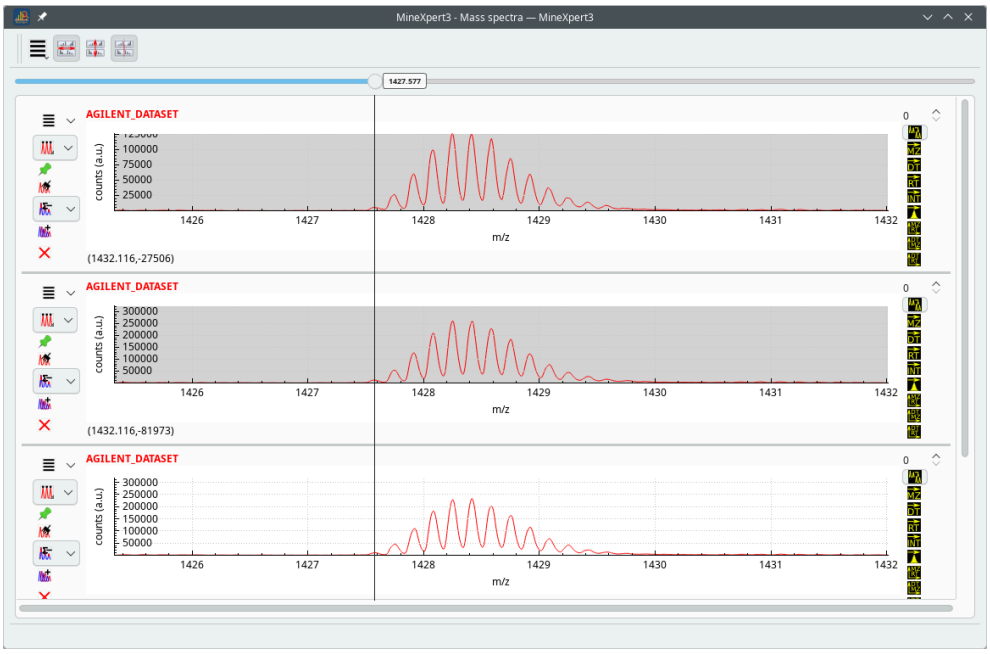


Figure 35: The vertical rule marker spans the whole window

When the X-axis is locked and the vertical ruler is shown, the abscissae value at which the rule is placed is displayed in a label.

4.8.4 General Plot Management in the Composite Plot Widgets

Each time a new plot is created in a composite plot widget, a menu item is made available in the Plot list submenu of the main menu. This is shown in Figure 36 – Plot management inside a composite plot widget.

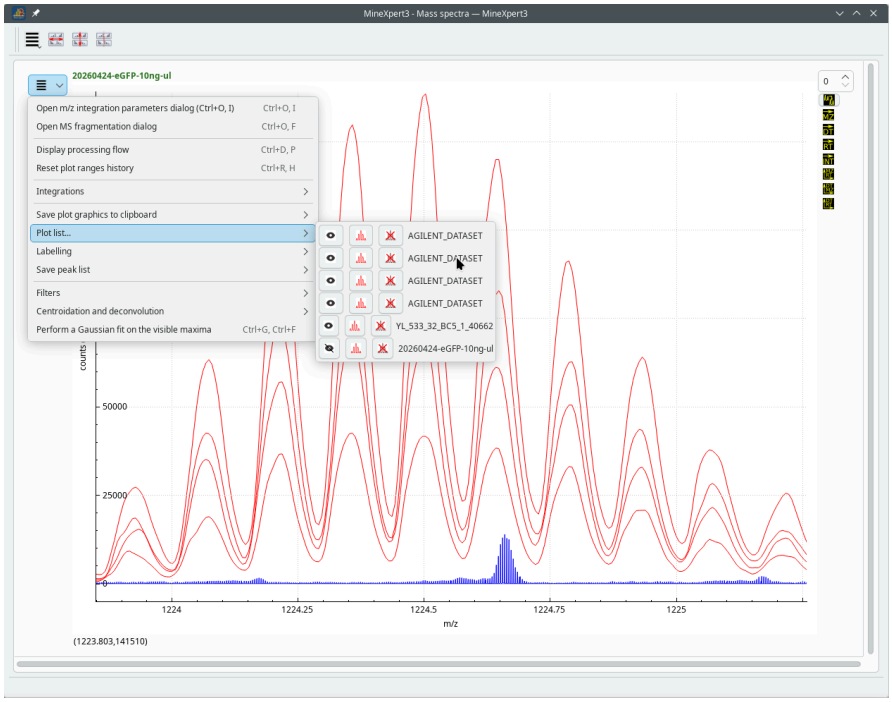


Figure 36: Plot management inside a composite plot widget

The three buttons displayed next to the plot name allow to show or hide the plot, to select or deselect it, or to destroy it.

5 Mass Data Integrations Featured by *MineXpert*

Analyzing mass spectrometric data (with or without drift data) usually involves performing various data integrations in sequence. We saw earlier that upon loading a mass spectrometry data file, the first data visualization that becomes available is the TIC chromatogram. Then, the user might ask to show the MS run data set table view (see Figure 9 – The table view-based scrutiny of one MS run data set). Thus, the TIC chromatogram and the table view are both starting points for the mass spectrometric data exploration. Once a number of mass data integrations have been performed, new integrations will be av ailable in *any* direction, by selecting the integration type in the plot widget (see Section 4.7.2 – The Right Column of Buttons Configures the New Integration to Run).

Regarding the parallel execution of mass spectral integrations, please see Section 2.5.3 – Using multiple threads during mass data integrations.

5.1 General Behaviour of Plot Widgets

There are patterns, in the way mass spectral data integrations can be configured, that are common to all plot widgets. These common patterns are described in the next sections. First, however, the *Processing Flow* concept needs to be described, as it serves as the basis for the integration configurations.

5.1.1 Processing Flow Entities Document all Integrations

Each time an integration is performed, *MineXpert* stores processing metadata that permits to characterize the integration for later internal reuse. Any integration, as it is performed, documents itself to the *Processing Flow*. For example, one integration might be from TIC chromatogram to mass spectrum, another integration can be from mass spectrum to ion mobility spectrum. These kinds of integration are documented as *Processing Types*. For each integration, there is thus a source data type and a destination data type. But this is not sufficient to document precisely the integration: there must be other metadata along with the integration source and destination data types. When integrating from TIC chromatogram to mass spectrum, the user might select only a region of interest in the TIC chromatogram. That is another kind of metadata that has to be documented: the source data range (in this case a retention time range). Likewise, when integrating from a mass spectrum to a drift spectrum, the origin is a *m/z* range and the destination is a drift spectrum. Other metadata are stored in the processing flow data, like the fragmentation

specification—if any has been set—and the *m/z* integration parameters, in case the integration produces a mass spectrum.

Together, all the processing flow metadata are stored in what are called *Processing Flow* instances. A processing flow instance can contain a default MS^n fragmentation specification and/or default *m/z* integration parameters. It can contain any number of *Processing Step* instances that in turn can contain any number of *Processing Spec* instances. Like for the processing flow, each processing step can contain a MS^n fragmentation specification and/or *m/z* integration parameters.

How is the processing flow documented? When a mass spectrometry data file is loaded from disk, *MineXpert* iterates in all the mass spectra of the MS run data set and computes a TIC chromatogram. That TIC chromatogram is displayed in the TIC/XIC chromatograms window. At this point, no MS^n fragmentation specification is set and no *m/z* integration parameters are set. When the user starts exploring the data, their first integration will determine how the processing flow will be filled-in with details about that integration.

If an integration is performed from the TIC chromatogram to a mass spectrum, then *m/z* integration parameters will necessarily be used. Either the user has configured these parameters (Section 5.1.3 – Setting the *m/z* Integration Parameters) and they will be used to perform the mass spectral combinations, or default values will be used (these default values are crafted from the statistical analysis of the whole MS run data set). The processing step that is created to document this specific integration will thus document the *m/z* integration parameters that have been used for the mass spectra combination. That processing step is added to the processing flow that documents the whole integration process. The plot that is created in the Mass spectra window will have, associated to it, that processing flow. Think of the *Processing Flow* as a pedigree ID card that each plot has associated to it. That pedigree gets incremented with new processing steps each time an integration is chained to the previous integrations.



Tip

The original feature of the *Processing Flow* concept is that, because each plot has its own processing flow, when a new integration is performed starting from that plot, then all the preceding steps of the processing flow are replayed and the new step is added to the processing flow. This has the beneficial effect of virtually restricting the scope of the initial data set into a much smaller data set all along the various integrations that occur during a data exploration session. When the scope reduces, the computing times reduce accordingly. However, the initial data set is not reduced in memory,

only the scope of it is reduced. This is much different from other software in which mass data are effectively pruned off the initial data set with the drawback that data must be reloaded each time an integration must start from the initial data set.

See Section 9 – The Processing Flow in a Data Exploration Session for a detailed description of the *Processing flow* concept.

5.1.2 Setting the MS^n Fragmentation Parameters

At any moment is it possible to set the MS^n integration parameters by selecting the *Open MS fragmentation dialog* menu item from the plot widget main menu (Section 4.7.3 – The Composite Plot Widget Main Menu).

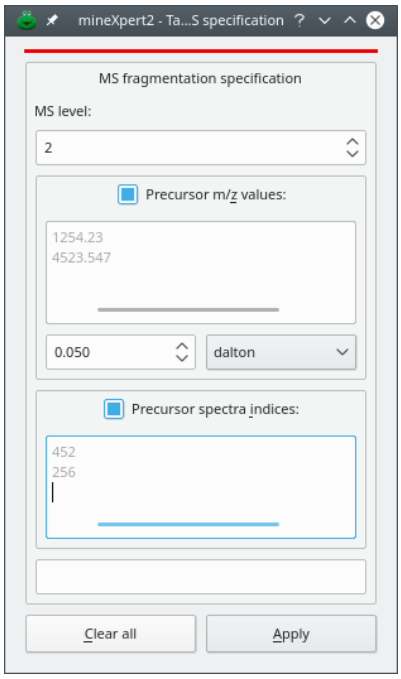


Figure 37: Setting the MS^n fragmentation parameters

The MS^n fragmentation parameters might be set at any time from the plot widget's main menu. The criteria are of three kinds, as described in detail in the text.

The dialog window that is displayed is depicted in Figure 37 – Setting the MS^n fragmentation parameters. The available settings are explained below:

- MS level: MS level that is targeted by the integration. For example, from a TIC chromatogram, setting MS level to 2 would perform an integration only accounting for mass spectra acquired for MS fragmentations of MS level 2, that is, for MS/MS spectra.

> **Note**
>
> Setting the MS level to *0* will perform an integration that takes into account *all* the mass spectrometric data, whatever their MS^n level.

- Precursor #mz; values: `Space` - or `RETURN` -separated list of *m/z* values. The precursor ions' *m/z* values are taken into account for the integration. When iterating in the mass spectra acquired in the MS run data set, each spectrum will be checked for the existence of an ion selection list. If any of the *m/z* values entered in this dialog window are found, the spectrum is accounted for; otherwise it is dismissed. A precursor *m/z* value is the *m/z* value of a precursor ion that was selected for fragmentation in a mass spectrum acquired right before the acquisition of the fragmentation spectrum.

> **Warning**
>
> There is a great probability that entering *m/z* values in the text widget above will fail to provide any result without some value tolerance set. Indeed, the matches are performed strictly using the entered values and the values recorded by the mass spectrometer, which might have 10 decimals! It is thus necessary to enter a tolerance value for the *m/z* value matches. In the figure, the tolerance is set to *0.05* dalton. Other "units" are available: resolution power and ppm (par-per-million).

- Precursor spectra indices: `Space` - or `RETURN` -separated list of integer values. The precursor spectra indices are taken into account for the integration. When iterating in the mass spectra acquired in the MS run data set, each spectrum will be checked for the existence of precursor spectra indices. If the precursor spectra indices entered in this dialog window are found, the spectrum is accounted for; otherwise it is dismissed. A precursor spectrum index is the number of the spectrum in the MS run data set that contained an ion that was selected for further fragmentation.

> **Note**
>
> Note that in the dialog window above, none of the three criteria is essential. Setting the MS level to *0* disactivates that parameter as a valid criterion for filtering spectra during the integration. In this case, only MS^n fragmentation specifications used in preceding

integrations are taken into account (that is, fragmentation specifications present in the *Processing Flow*). If one wants to reset the MS^n fragmentation specification to a lower MS level than that of the last preceding integration, the only way to achieve this is to go back to the previous plot where the processing flow contained processing steps of the right MS^n fragmentation level values. Indeed, it is not possible to reset the MS^n fragmentation level in a plot by setting the `MS level` to *0* because that plot already has a processing flow containing processing steps configured with a non modifiable MS level.

There is a shortcut to the MS^n fragmentation MS level in each composite plot widget, as shown below, where the spin box widget on top of the right hand side column of allows one to set the MS level.

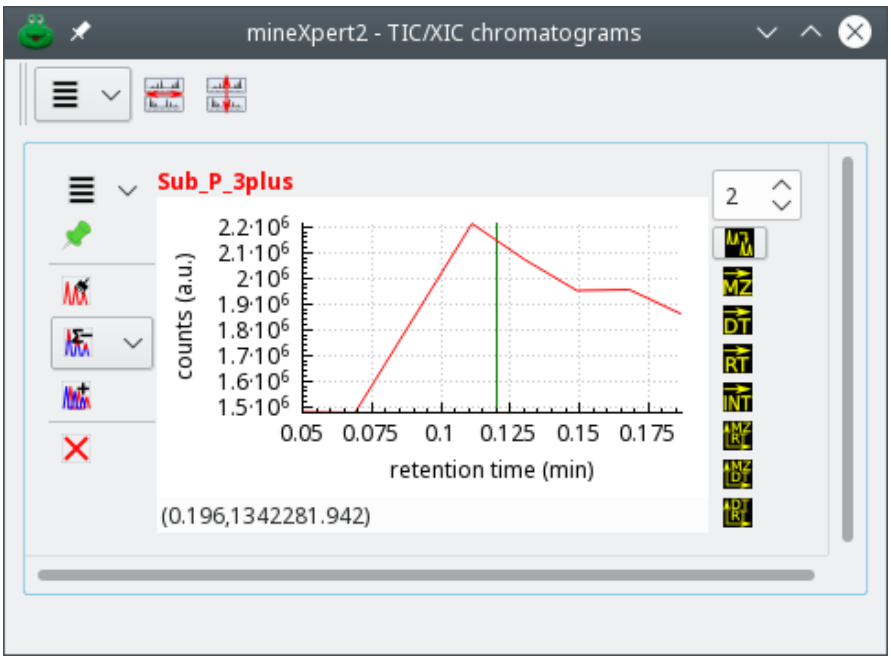


Figure 38: Shortcut to the fragmentation MS level setting

If the only fragmentation setting to be modified is the MS level, then, instead of opening the dialog window described above, it is possible to set the required MS level right into the spin box widget at the top of the right hand side column of buttons in the composite plot widget.

5.1.3 Setting the *m/z* Integration Parameters

The *m/z* integration parameters are only compulsorily used in case an integration will produce a mass spectrum or a colormap involving *m/z* data in any one of the two

dimensions. Indeed, these parameters are required to configure the way mass spectra are combined together into a single product mass spectrum.

An integration to a mass spectrum occurs when the user 🖱-selects a range in a given plot while the MZ checkbutton is checked. Integrations to a mass spectrum can be elicited from any plot.

Very rarely can one perform an integration to a mass spectrum without having to first configure the way the *m/z* values are binned into the combination spectrum. The determination of the best *m/z* integration parameters is a process that might be required each time a new data set comes from an instrument from which no data have been obtained before. Indeed, there is a vast number of different mass spectral data acquisition/storage modes that depend on the vendors or on the mass spectrometer models. In general, this diversity of mass spectral data files[9] creates difficulties in the mass spectra integrations, as shown below in Figure 39 – Unusable combination spectrum without binning.

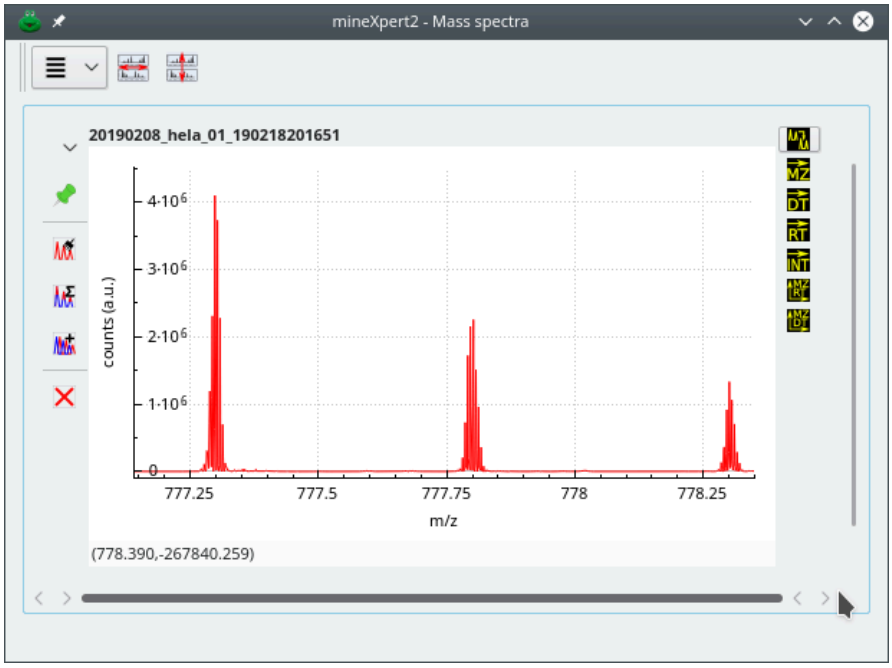


Figure 39: Unusable combination spectrum without binning

The mass data used to compute this combination spectrum originate from a Lumos Orbitrap analyzer. The visible signal should have been three peaks belonging to a 2-charged ion isotopic cluster. Each peak of the isotopic cluster is artifactually made of numerous data points because the data from the Orbitrap analyzer have 10 decimal digits. Without binning, the mass spectrum is almost unusable.

In this combination mass spectrum, computed without binning from Lumos Orbitrap-originating data, three peaks that belong to an isotopic cluster appear to be made from a large number of "sub-peaks". This is due to the fact that the number of decimals for

[9] Even if they are obtained by conversion of raw vendor files to mzML files using *ProteoWizard*'s *msconvert* tool.

the *m/z* values in the data file is so high that a single isotopic peak appears as a large set of sub-peaks. While the isotopic cluster's peaks are "readable" below their shape, the signal in this mass spectrum is almost totally useless. This is a perfect illustration of the necessity of data binning in the combination spectrum when integrating mass spectral data to a mass spectrum.

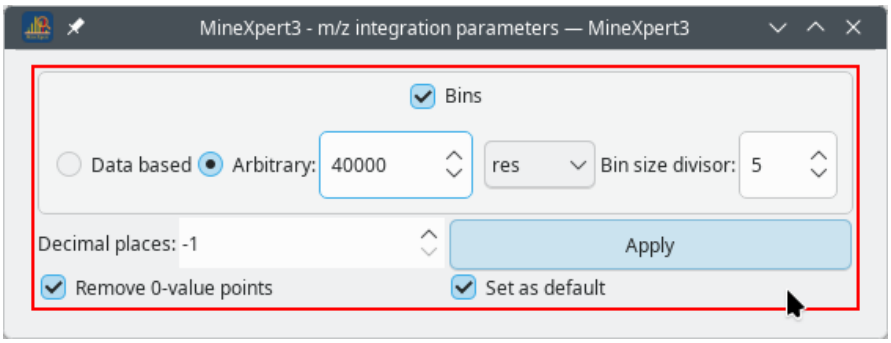


Figure 40: The m/z integration parameters window

Integrations to any plot that involves *m/z* as one of its dimension (be that a mass spectrum or a colormap) can be configured to ensure the best results, depending on the kind of mass data. Proper binning configuration is key to getting best results.

MineXpert provides a number of ways to configure mass spectral combinations such that the obtained mass spectrum is usable. The *m/z* integration parameters that might be set are described in the following sections. The window depicted in Figure 40 – The m/z integration parameters window can be displayed from any plot widget menu using the *Open the m/z integration params dialog* menu item (see Section 4.7.3 – The Composite Plot Widget Main Menu).

- If no binning is required, just make sure the Bins check box is unchecked. Data coming from some *Agilent* or *Waters* ion mobility mass spectrometers are very well binned without need to configure binning in *MassXpert*.
- If binning is required, make sure the Bins check box is checked. This is always the case with data coming from *ThermoFisher Orbitrap*-based instruments or *Bruker* instruments. Next, set the right parameters:
  - Data based binning means that the binning will be set according to the current data;
  - Arbitrary binning means that the size of bins is set arbitrarily to a value of which the unit might be res (resolution), dalton or ppm (part-per-million).
  - Bin size divisor is a factor by which the bin size is further reduced after having been computed on the basis of the Arbitrary settings.

- Decimal places: when performing the combination, each *m/z* value needs to be restricted to that number of decimal places. This setting may in some situations improve the signal or accelerate the speed of the combination for very large datasets.

> **Note**
>
> If Decimal places is let to the default value −1, that datum will be determined experimentally by looking into the bins size. Typically, if the bins size is, for example, 0.05, then, the decimal places will be determined to be 2 (number of 0 decimals after the decimal point, plus one). If the bins size is 0.003, the decimal places will be determined to be 3.

- Remove 0-value data points: remove all the (*m/z*, i) pairs having a naught i value.

> **Tip**
>
> If the settings are regularly used during the data visualization and exploration sessions, checking the Set as default check box will ensure that these settings are applied by default in all the future integrations to a mass spectrum.

When all the parameters have been set, click onto Apply button or key in CTRL RETURN. At this point the next integrations started from the plot widget in which these *m/z* integration parameters were set will be performed using the settings. These parameters will propagate to all the descendant widgets. It will be possible, at any moment and in any widget, to modify the parameters again.

5.1.4 Step-by-step Determination of the Best *m/z* Integration Parameters

This section provides guidelines on the best way to determine the *m/z* integration parameters that will result in nicely shaped mass peaks in the combination mass spectrum. There might be situations where there is no need to set the *m/z* integration parameters, typically if the mass data have been properly binned by the software running the mass spectral data acquisition (*Waters* and *Agilent* systems often provide nicely binned mass spectra).

However, in a vast majority of the cases, proper setting of the *m/z* integration parameters will be essential to achieve a combination of mass spectra resulting in a useful mass spectrum. In particular, it is important to grasp that the binning might need to be dynamically set while combining (*m/z*, i) pairs along the *m/z* axis. This dynamic binning is configured

by choosing either res or ppm (for resolving power or part-per-million, respectively; bins have varying sizes across the whole *m/z* range).

When dealing with a mass spectrometry data set from an instrument that is unknown, the first step is to perform an integration to a mass spectrum without binning. In this situation, *all* the distinct *m/z* data points observed in all the mass spectra being combined together are going to be stored in the combination spectrum. When the data come from an instrument that does not produce nicely binned mass spectra, the resulting combination mass spectrum might be highly spiky, with spurious data points interspersed between seemingly correct data points. This is typically visible in the top mass spectrum shown in Figure 41 – Determination of the best *m/z* integration parameters for integration to mass spectra.

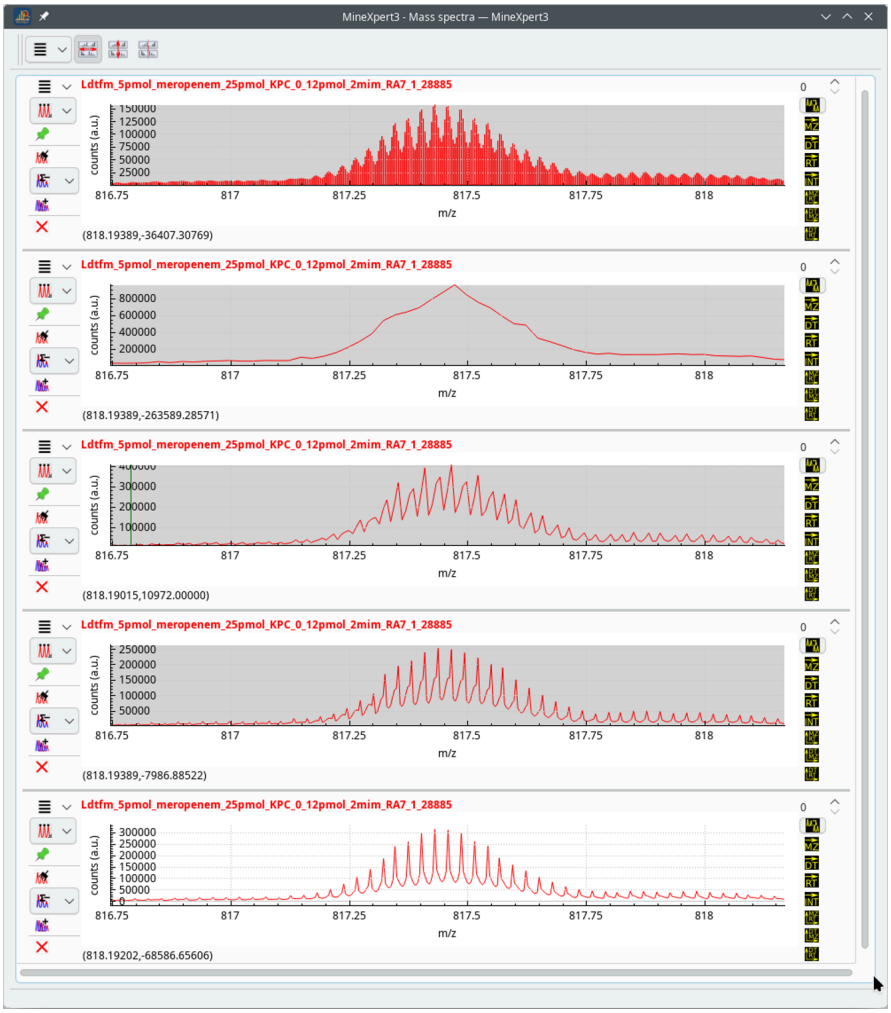


Figure 41: Determination of the best *m/z* integration parameters for integration to mass spectra

Zoomed in view of a protein mass spectrum combined with differing *m/z* integration parameters.

The different plots in the figure show the mass spectral data combined for the same acquisition data and for the same retention time range. From top to bottom:

- *No binning*. Visibly, each peak that is part of the isotopic cluster is made of a number of peaks that appear to be superfluous.
- *Bins res: 30000, bin size divisor: 1*. This resolving power was used as a starting value for exploring the data because the apparent resolving power was found to be around that value in Figure 42 – Estimating the resolving power of a mass spectral acquisition. Remarkably, the bin size divisor value of 1 cannot permit to shape the isotopic cluster peaks properly.
- *Bins res: 30000, bin size divisor: 5*. Dividing the bin size determined above for resolving power of 30000 starts to give meaninful results: the peaks of the isotopologues start to be shaped. They are still a bit rough.
- *Bins res: 30000, bin size divisor: 6*. Using value 6 instead of 5 refines the isotopologue peak shapes. However, there is still an artifactual shoulder on the left of the isotopologue peaks.
- *Bins res: 40000, bin size divisor: 5*. That is the right combination: resolving power of 40000, with bin size divisor of 5. The charge determination by deconvolution of the isotopic cluster yields the right charge of the polymer: +36.

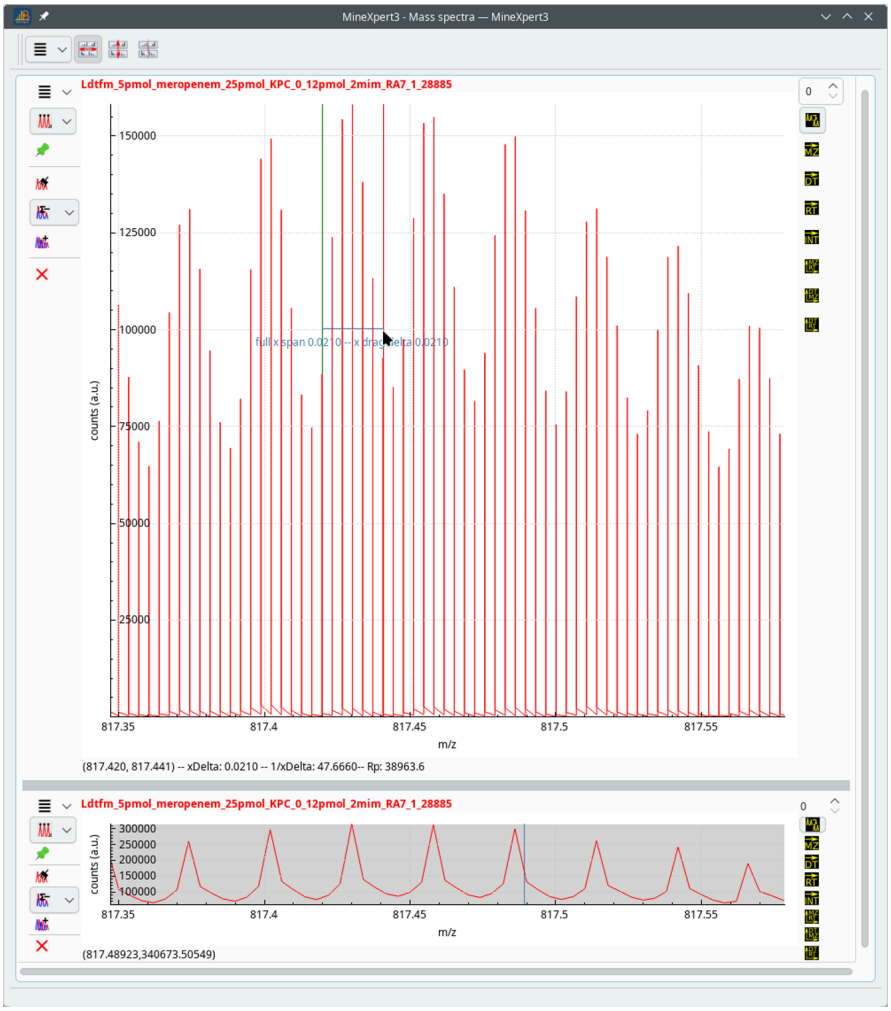


Figure 42: Estimating the resolving power of a mass spectral acquisition

Dragging the mouse cursor between the extremities of an isotopic cluster peak, at half of its height, provides a rough estimation of the resolving power of the instrument.

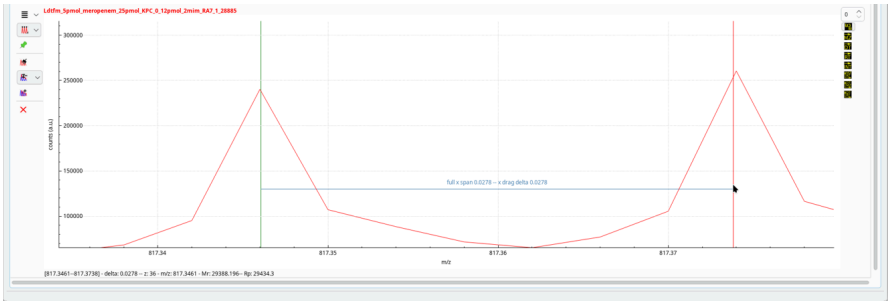


Figure 43: Determination of the ion charge by using the isotopic cluster

One evidence that the setting for this integration is correct is that it is now possible to use the inter-peak deconvolution method for the determination of the ion charge (36+), which confirms the result of a previous deconvolution based on the charge state envelope.

One important concept to grasp is that the bin size used for the mass spectra combinations will actually determine the final resolution of the mass spectrum (or colormap). For

example, imagine that the setting is the following: Arbitrary binning with bin size of *1 Da* in that case the mass spectrum or the *m/z*-based colormap will be very rough. Instead, setting the bin size to 0.05 will provide a much better spectrum (or colormap). Of course, going down with the size of the bins will increase almost linearly the computation time required to go through them all while adding signal from the various spectra that are being combined. This is exemplified in Figure 44 – Effects of the bin size on the colormap involving one *M/Z* dimension.

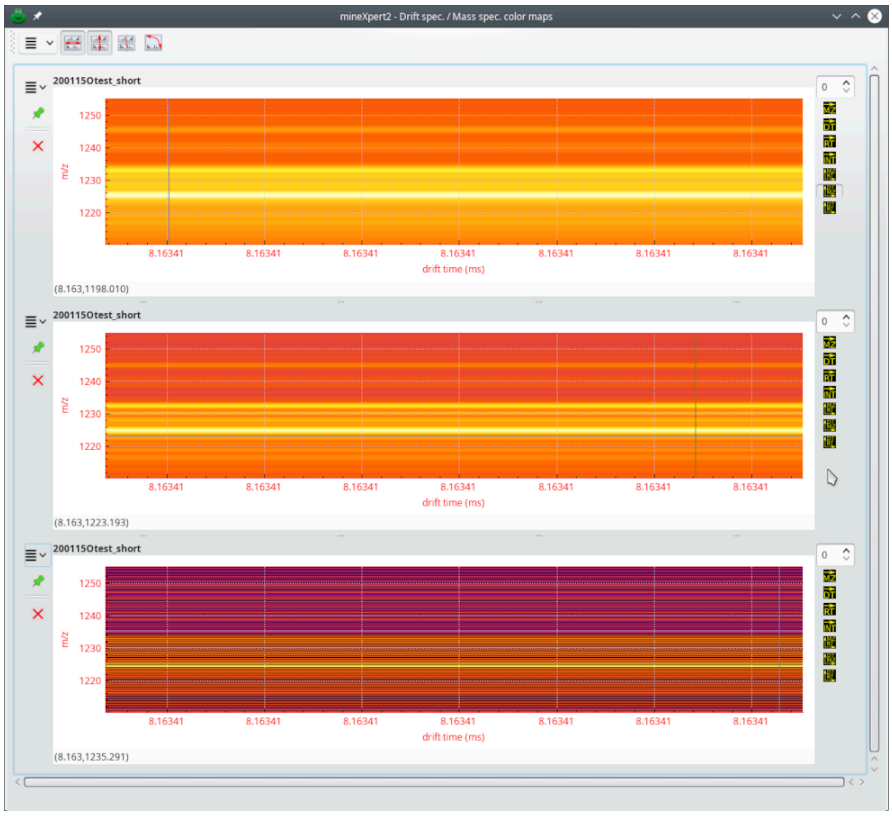


Figure 44: Effects of the bin size on the colormap involving one *M/Z* dimension

The three different colormap regions were computed in the Arbitrary binning mode with bin sizes 1 Da, 0.1 Da and 0.01 Da (top to bottom).



Tip

Because the computation time increases inversely with the size of the bins, it is recommended to start a colormap-based data exploration session with an integration over the whole data set but with *large* bins (typically 1 Da) that will provide enough sight on the data. Then, when the feature of interest has been located on the colormap to a given region, reduce the bin size to increase the resolution of the colormap and integrate from that same region to a new colormap.

5.1.5 Removing 0-intensity *m/z* data points is useful

Mass spectrum data points with an intensity value equal to 0 might arise either from the initial data, as read from the file, or from the binning process, as detailed below.

When bins are prepared, prior to starting a mass spectral combination operation, they all correspond to a *m/z* value associated to a 0-value intensity. Once the bins have been prepared, the combination starts and, while iterating through all the data points of all the spectra, fills-in the bins. If, however, a bin is never filled (it ends up never having been updated during the combination), that bin will have an intensity of 0. Bins having a 0-intensity value have a bad effect on the result mass spectrum. Removing them by setting the Remove 0-value data points parameter proves beneficial, as shown in Figure 45 – Removing 0-intensity data points. When the 0-intensity data points are not removed (upper spectrum), the signal is deteriorated by spurious inverted spikes. Removal of the 0-intensity data points, cleans the trace perfectly.

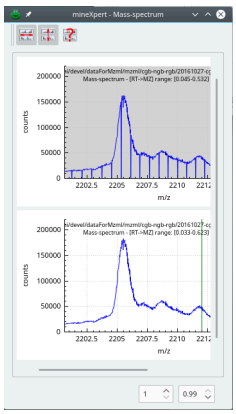


Figure 45: Removing 0-intensity data points

When arbitrary binning is performed, residual 0-intensity data points might survive in the combination spectrum, which deteriorates the resulting mass spectrum. Removing these data points from the combined mass spectrum cleans the trace.

5.1.6 Savitzky-Golay Filtering of Any Kind of Data

The Savitzky-Golay filtering method is widely known for its effectiveness in removing noise from mass spectral data.

It is possible to apply that filter to any plot by selecting the corresponding menus as described at Figure 46 – Menus to configure and run the Savitzky-Golay filter to smooth traces. The Savitzky-Golay parameters can be set by selecting the Configure Savitzky-

Golay smoothing filter menu as shown on Figure 46 – Menus to configure and run the Savitzky-Golay filter to smooth traces.

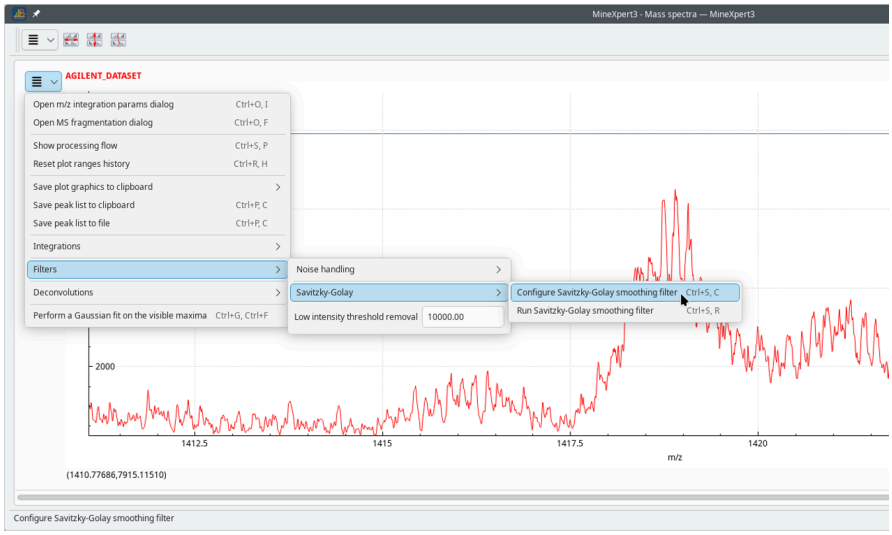


Figure 46: Menus to configure and run the Savitzky-Golay filter to smooth traces

The Savitzky-Golay algorithm is a powerful filter that allows one to smooth any graph. The first menu enables the configuration of the filter and the second menu actually applies the filter to the trace.

The Savitzky-Golay filter requires careful configuration. The configuration provided by default is shown in Figure 47 – Configuration of the Savitzky-Golay filter to smooth traces.

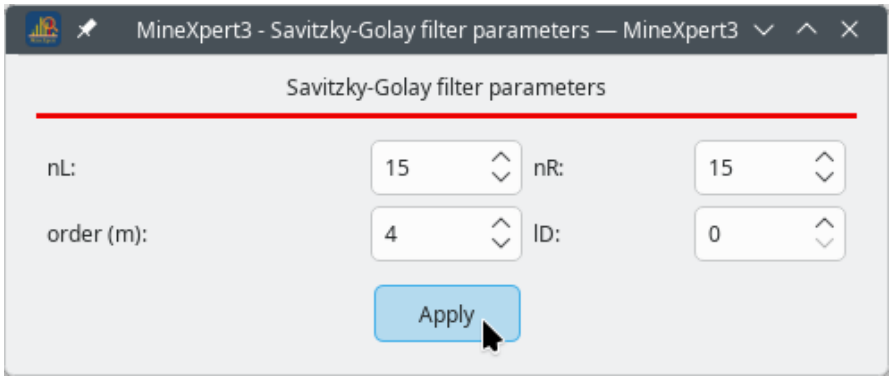


Figure 47: Configuration of the Savitzky-Golay filter to smooth traces

The different parameters used to configure the Savitzky-Golay algorithm are described in the text.

- nL: specifies the number of data points to the left of the point being filtered;
- nR: specifies the number of data points to the right of the point being filtered. The total number of points in the window that is considered for the regression is thus nL + nR + 1;
- m: specifies the order of the polynomial to use in the regression analysis leading to the Savitzky-Golay coefficients (typically between 2 and 6);

- ID: specifies the order of the derivative to extract from the Savitzky-Golay smoothing algorithm (for regular smoothing, use value 0);

5.2 Various Mass Spectral Data Integrations

Now that the configuration patterns common to all plot widgets have been described, the various mass spectral data integrations can be reviewed.

> **Tip**
>
> All the integration operations are performed using the *right* mouse button (🖱). This setting allows to easily distinguish all these integration operations from all the 🖱-based visualization operations (Section 4.8 – General Operation of the Plot Widgets).

- *Integrations to a mass spectrum* This kind of operation is triggered upon 🖱-click-dragging the mouse over the region of interest after having selected the MZ button on the right button column of the widget. It is also possible to avoid clicking onto the button by pressing the M keyboard key while performing the 🖱-click-drag movement. *MineXpert* integrates all the spectra that have been acquired between the start marker and the end marker as set during the mouse drag movement. A new mass spectrum is then plotted in the Mass spectra window.



Figure 48: Integrating mass data from a TIC chromatogram to a mass spectrum

The integration is defined to be a *[TIC chromatogram to mass spectrum]* integration because the plot widget is in the TIC/XIC chromatograms window and the MZ push button is checked.

As seen on the figure above, the region defined by the 🖱-click-dragging operation is delimited by arrows, a green marker at the start and a red marker at the end. The arrows at the ends of the horizontal line allow one to make the difference between an integration

selection and a simple distance-measuring selection (detailed at Section 7 – Mass Spectral Deconvolutions).

- *Integrations to a drift spectrum* This kind of operation is similar to the one described above, unless for the DT button that is now checked. It is also possible to avoid clicking onto the button by pressing the D keyboard key while performing the 🖱-click-drag movement. The new plot is added to the Drift spectra window.
- *Integrations to a XIC chromatogram* This kind of operation is similar to the one described above, unless for the RT button that is now checked. It is also possible to avoid clicking onto the button by pressing the R keyboard key while performing the 🖱-click-drag movement. The new plot is added to the TIC/XIC chromatograms window.
- *Integrations to TIC intensity using the processing flow* This operation involves 🖱-selecting a plot region of interest encompassing the mass spectral feature for which the intensity is to be integrated while the INT button is checked. It is also possible to avoid clicking onto the button by pressing the I keyboard key while performing the 🖱-click-drag movement. The obtained numerical result is displayed in the status bar below the plot widget and in the Console window.

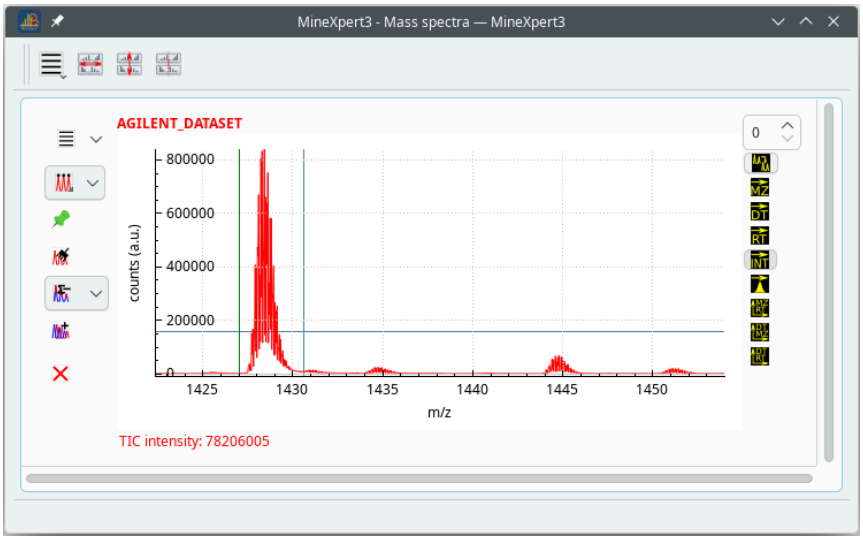


Figure 49: Integrating mass data from a mass spectrum to a single TIC intensity value using the processing flow

The integration is defined to be a *Mass spectrum to TIC intensity* integration because the plot widget is in the Mass spectra window and the INT push button is checked. The TIC intensity value is printed in the status bar.

- *Integrations to TIC intensity using the plot data points* This operation involves 🖱-selecting a plot region of interest encompassing the mass spectral feature for which the

intensity is to be integrated while the is checked. It is also possible to avoid clicking onto the button by pressing the Y keyboard key while performing the -click-drag movement. The obtained numerical result is displayed in the status bar below the plot widget and in the Console window.

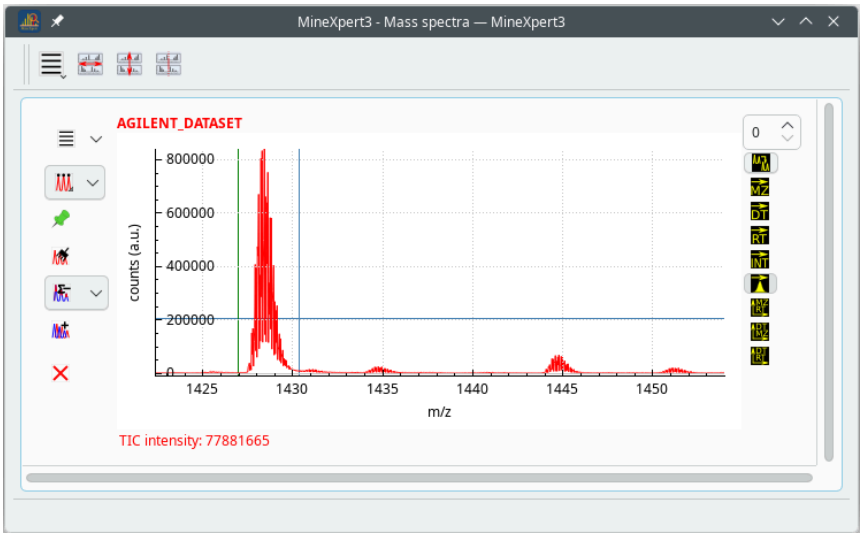


Figure 50: Integrating mass data from a mass spectrum to a single TIC intensity value using the plot data points

The integration is defined to be a *Mass spectrum to TIC intensity* integration because the plot widget is in the Mass spectra window and the push button is checked. The TIC intensity value is printed in the status bar.

Tip

The integration to a TIC intensity value using the datapoints (destination is) is useful when one wishes to know the intensity of a signal from a plot that is logically detached from the initial dataset loaded from disk. This is explained in detail at Section 9.2 – Recording Processing Steps Detached from the Initial Mass Spectrometry Dataset and use case examples are provided at Section 7.2.2 – Multi-point statistical analysis-based charge determination and at Section 7.2.3 – Multi-point statistical analysis-based charge determination and mass estimations.

- The same mechanics is at work in the other plot widget windows. For example, to trigger the integration of any kind of plot to an intensity = f(dt,*m/z*) colormap, simply check the button and drag the mouse over the plot region of interest.

5.2.1 Considerations on the Diversity of Mass Data Contents

Loading data from mass data files in mzML format does not guarantee that the data will be of the same kind when they originate from different mass spectrometers. For example, data from Orbitrap mass spectrometers have the following characteristics:

- All spectra do not start at the same *m/z* value;
- All spectra do not have the same number of data points (they do not have the same size);
- A large number of data points might have 0 intensity values (intensity at a given *m/z* value is 0);
- The *m/z* delta between two consecutive *m/z* values is not constant, and this is the major difficulty for data integration to a mass spectrum.

This is the output of the statistical analysis of the data loaded from a Lumos Orbitrap-originating file:

```
Spectral data set statistics:
Total number of spectra: 6203
Average of spectrum size: 391.311946
StdDev of spectrum size: 168.062934
Mininum m/z value: 400.007111
Average of first m/z value: 401.448935
StdDev of first m/z value: 1.590049
Maximum m/z value: 1999.928589
Average of last m/z value: 1901.852315
StdDev of last m/z value: 45.864131
Minimum m/z shift: -0.344452
Maximum m/z shift: 0.000000
Average of m/z shift: 1.097372
StdDev of m/z shift: 1.590049
Smallest Delta of m/z (step): 0.006195
Average of smallest Delta of m/z (step): 0.023757
StdDev of smallest Delta of m/z (step): 0.013179
Greatest Delta of m/z (step): 405.356934
Average of greatest Delta of m/z (step): 163.112057
StdDev of greatest Delta of m/z (step): 75.947334
```

As mentioned earlier, the most interesting bit of information is in the line reproduced below:

```
Smallest Delta of m/z (step): 0.006195
```

That 0.0062 value somehow gives an indication of the "definition" of the spectrum, that is, the smallest distance possible between two consecutive points in the *m/z*-axis.

In general, the fact that the spectra of an acquisition do not all have the same *m/z* vector as the *m/z*-axis is a great difficulty for mass spectral integration because it requires setting up binning prior to performing the mass spectral combination. That binning is nothing else than crafting a *m/z* value vector able to receive the intensities of all the *m/z* data points in the spectra to be combined. These concepts are developed in the next paragraph and have also been reviewed at Section 5.1.3 – Setting the *m/z* Integration Parameters.

5.2.2 Statistical Analysis of Mass Data

At the end of the data file loading, *MineXpert* performs a rudimentary statistical analysis of the data. The main datum of interest is the smallest *m/z* step that is observed in the whole set of mass data loaded from disk (the mass spectrum list, that can hold mass spectra in the tens of thousands). For each mass spectrum in the list, the smallest *m/z* delta between any two consecutive data points is recorded. Then, the smallest ever *m/z* delta value is sought amidst all the recorded values. Intuitively, that smallest *m/z* delta value provides an idea of the resolution power of the instrument that generated the mass spectra. Interestingly, this is not the proper value to configure binning. The best value is the median value of the smallest *m/z* delta values encountered over all the mass spectra of the data file. It is the value that is suggested by default to arbitrarily construct the bins during an integration to a mass spectrum, as described in Figure 40 – The m/z integration parameters window (Arbitrary binning value with bin size unit dalton).

5.3 Integrations Originating in Color Map Plots

There are two different kinds of plots: the graph or bars plots (spectra, centroids,for example) and the "colormap plots". TIC chromatograms, mass spectra and ion mobility spectra are trace plots. Colormap plots do plot intensity values as colors for pairs of data, like mass spectra as a function of retention times or ion mobility, or retention times as a function of ion mobility.

When an integration starts from a graph plot, the integration region selection (with a right-button mouse drag operation) of a given plot range (over the X axis) can be performed without drawing a rectangle over the region of interest. Indeed, only the X axis is the criterion for the selection of the region of interest. When an integration starts from a colormap plot, the second dimension (the Y axis) has a meaning for the selection of the region of interest. In colormap plots, the selection region has to be bi-dimensional, that is, not a horizontal line, but a real rectangle over both the X and the Y axes.

By default, the rectangle that is drawn upon mouse dragging is a real rectangle, that is, a four-square-corner polygon. There are a number of situations, however, where the user might want to select a non-rectangle region, like if the rectangle were skewed. This is illustrated in Figure 52 – Integrations from a colormap plot using a skewed selection rectangle (last step)

Drawing a skewed selection rectangle is a two-step process. One first draws the base of the rectangle, as usual, with a horizontal-only mouse drag movement (see Figure 51 – Integrations from a colormap plot using a skewed selection rectangle (first step)). When the base of the rectangle has the right width, the user presses the `S` keyboard key. Pressing that keyboard key triggers the definition of the width of the skewed selection rectangle, that gets stored in memory.

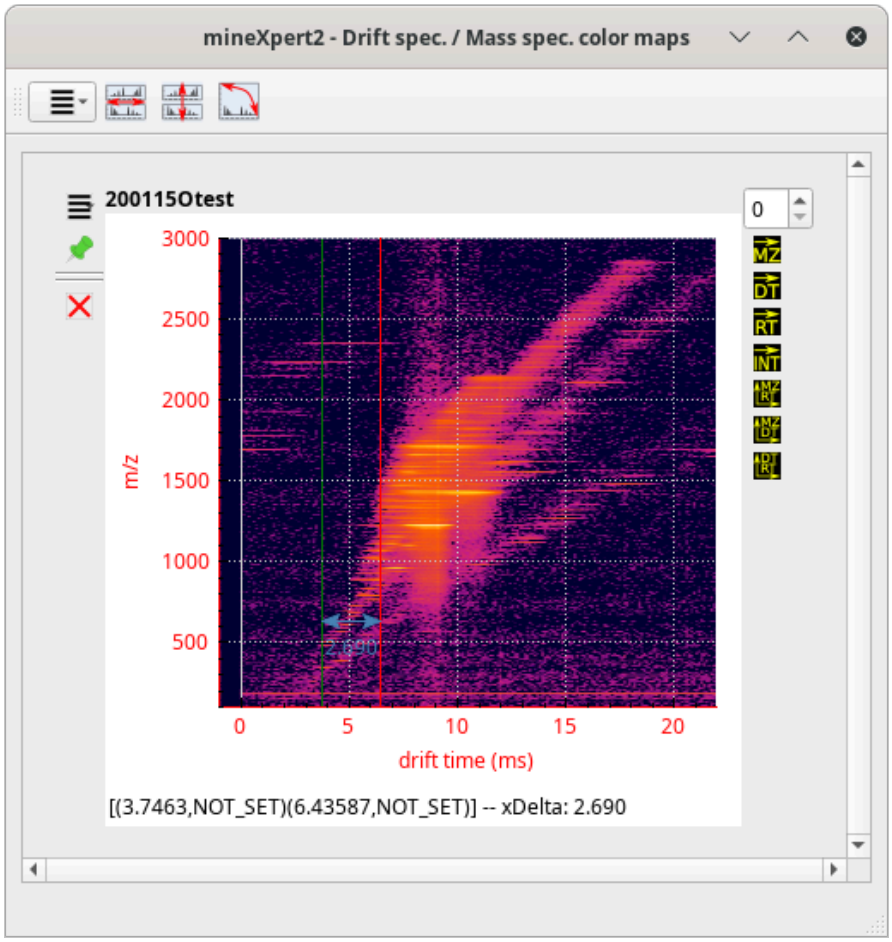


Figure 51: Integrations from a colormap plot using a skewed selection rectangle (first step)

> The first step in the definition of a skewed selection rectangle is to right-button-drag the mouse along the X axis in order to draw a selection line that has the final skewed rectangle width. Once the width is properly defined, the user presses the `S` keyboard key, which stores the rectangle width in memory.

When the mouse drag operation draws the actual two-dimensional polygon (by dragging the mouse cursor also along the Y axis), the user maintains the `Alt` keyboard key

pressed. Maintaing that key pressed has the effect of switching the drawing of a conventional four-square-corner selection rectangle into a skewed selection rectangle (see Figure 52 – Integrations from a colormap plot using a skewed selection rectangle (last step)).

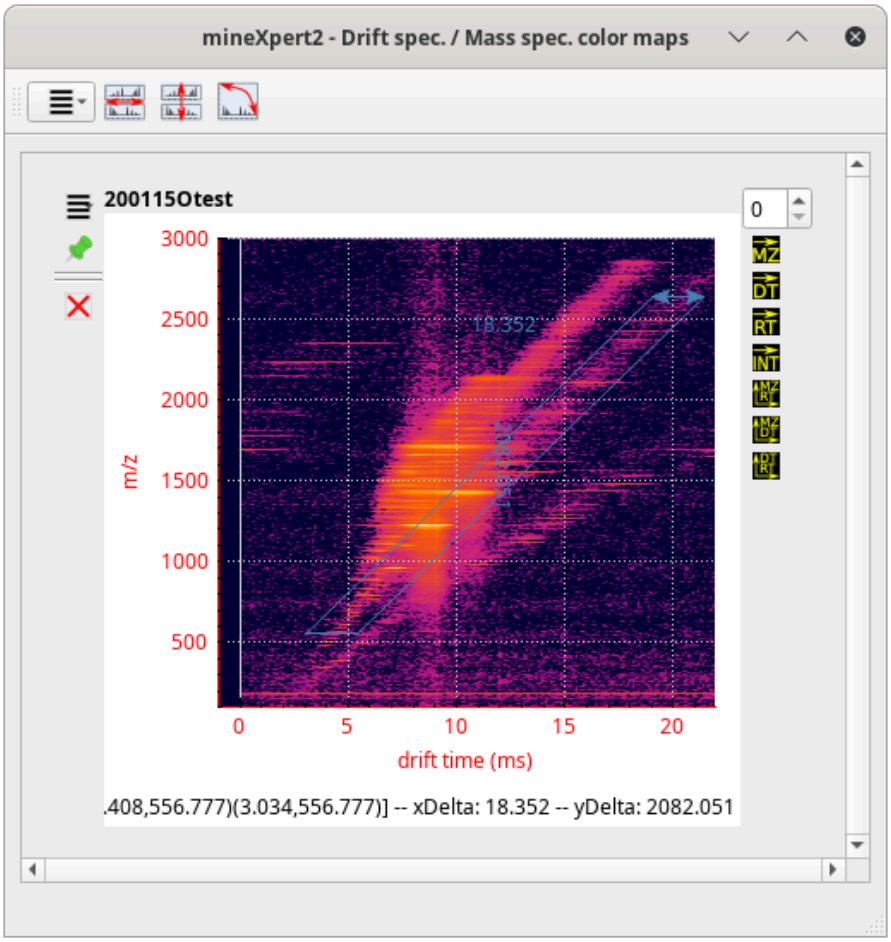


Figure 52: Integrations from a colormap plot using a skewed selection rectangle (last step)

The last step in the definition of a skewed selection rectangle is to drag the right mouse button over a bi-dimensional space (that is, also along the Y axis) while maintaining the `Alt` modifier key pressed. The rectangle that is drawn is now skewed, in contrast to a normal square-corner rectangle.

When the mouse right-button is released, the integration computation is started that involves checking if the points that are processed are indeed contained inside the skewed selection rectangle. The result is shown in Figure 53 – Integrations from a colormap plot using a skewed selection rectangle (result).

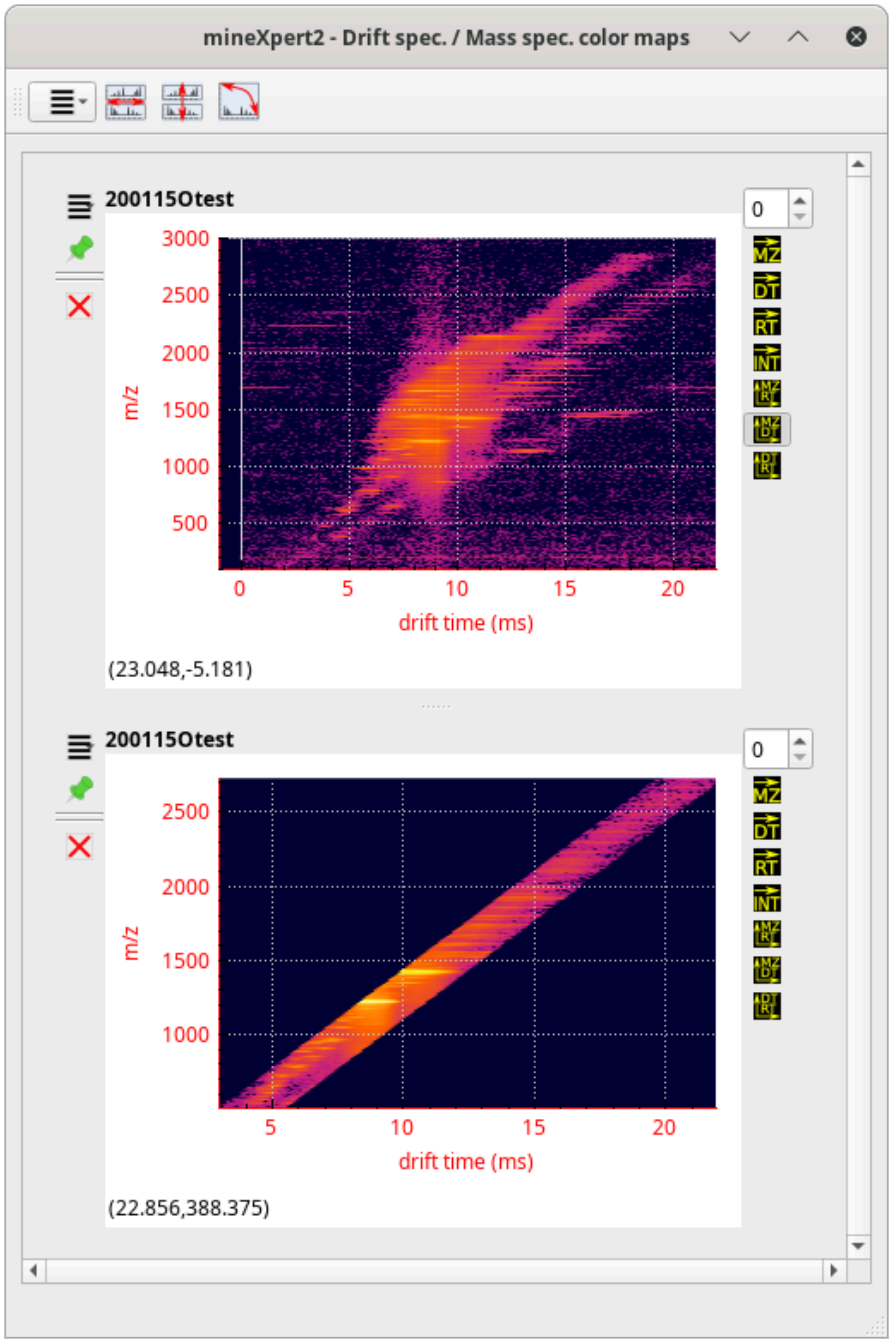


Figure 53: Integrations from a colormap plot using a skewed selection rectangle (result)

The result of a skewed selection rectangle-based integration is inherently of the shape of the selection.

5.4 Signal Enhancement and Noise Reduction in Color Map Plots

The processes described in this section have been mentioned at Section 4.7.3.2 – The Color Map Composite Plot Widget Main Menu.

The signal obtained in ion mobility–mass spectrometry experiments may be of too little dynamics, which hamper proper visualization of details. One solution is to run a Z axis

scale (the axis holding intensity values) conversion from linear scale to log10 scale (see Figure 31 – Colormap plot widget main menu). While this has the benefit of highlighting details, it also has the drawback of potently enhancing noise. As of version 8.1.0, *MineXpert* provides means to signal-to-noise ratio tweaking as detailed below.

There are two different approaches to signal-to-noise ratio enhancement:

- *Low-pass filter Z axis data*: The value that is entered in the spin box widget is a *percentage* of the maximum intensity over the whole map. The function first computes a threshold using the formula threshold = percentage * max_int / 100. All the data points having an intensity value above that threshold have their value set to that threshold value. All other data points are unchanged. This function is used to reduce the dynamic range of the map and potentially enhance features of interest.

- *High-pass filter Z axis data*: The value that is entered in the spin box widget is a *percentage* of the maximum intensity over the whole map. The function first computes a threshold using the formula threshold = percentage * max_int / 100. All the data points having an intensity value below that threshold have their value set to that threshold value. All other data points are unchanged. This function is typically used for removing noise. Indeed, by setting the percentage value to 5, for example, all the data points having a pretty low intensity value (that is, below 5 % of the maximum intensity value in the whole map data) will be set to that value. An example is provided in Figure 54 – Example of signal-to-noise ratio enhancement.

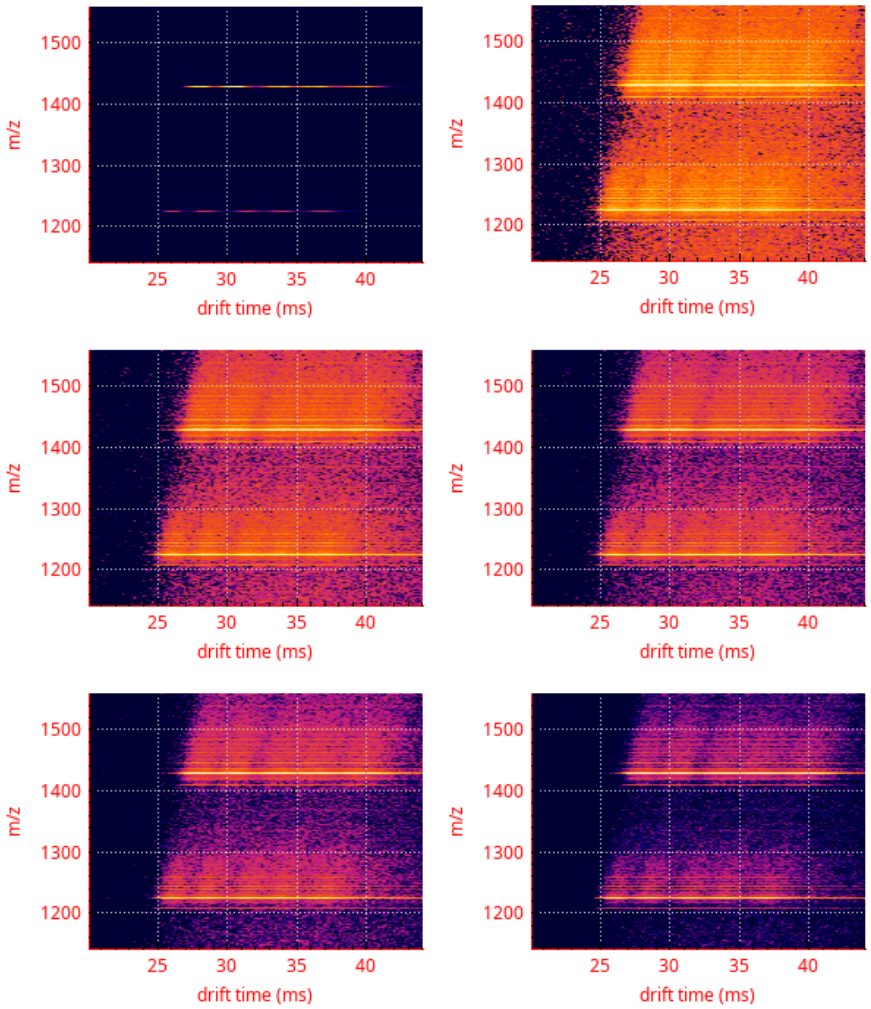


Figure 54: Example of signal-to-noise ratio enhancement

The top left vignette shows the colormap as it is computed. The top right vignette show that colormap after a log10 recomputation has been performed. The remaining vignettes correspond to the application of a high pass filter with percentage values increasing from 10 to 40 in 10 % increments.

5.5 Low Intensity Signal Removal in Traces

Any trace (that is, any plot that is not a colormap) can be noisy. That is particularly true for mass spectra coming from non-Orbitrap analyzers. If one can tell what is the *level* of the noise, it is possible to remove all the peaks that are below that level. The process is complex because it needs to take into account ever-more frequent high resolution mass spectrometry data that shows resolved isotopic cluster for which certains peaks might be below the threshold while others are above threshold. Of course, if any peak of an isotopic cluster is above threshold, then the whole isotopic cluster, no matter how it might comprise below-threshold peaks, should be retained.

At the time of this writing, the low intensity signal removal filter is working best for samples containing large analytes, where the isotopic cluster is most approaching an overall Gaussian shape. It is considered that this feature is still experimental as it still has a number of rough edges. The step are described in the following figures (see Figure 56 – Example of low intensity signal removal (configuration)).

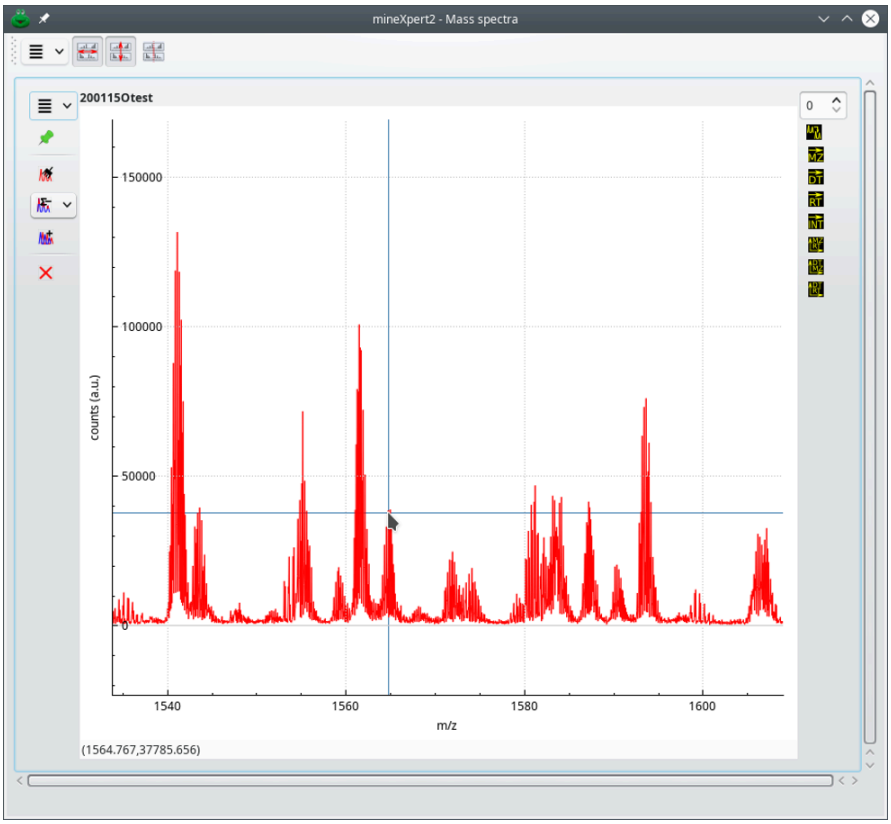


Figure 55: Example of low intensity signal removal (starting trace)

This is a segment of the initial mass spectrum with some isotopic clusters having part of their peaks above the 50000 threshold and some others that are entirely below that threshold.

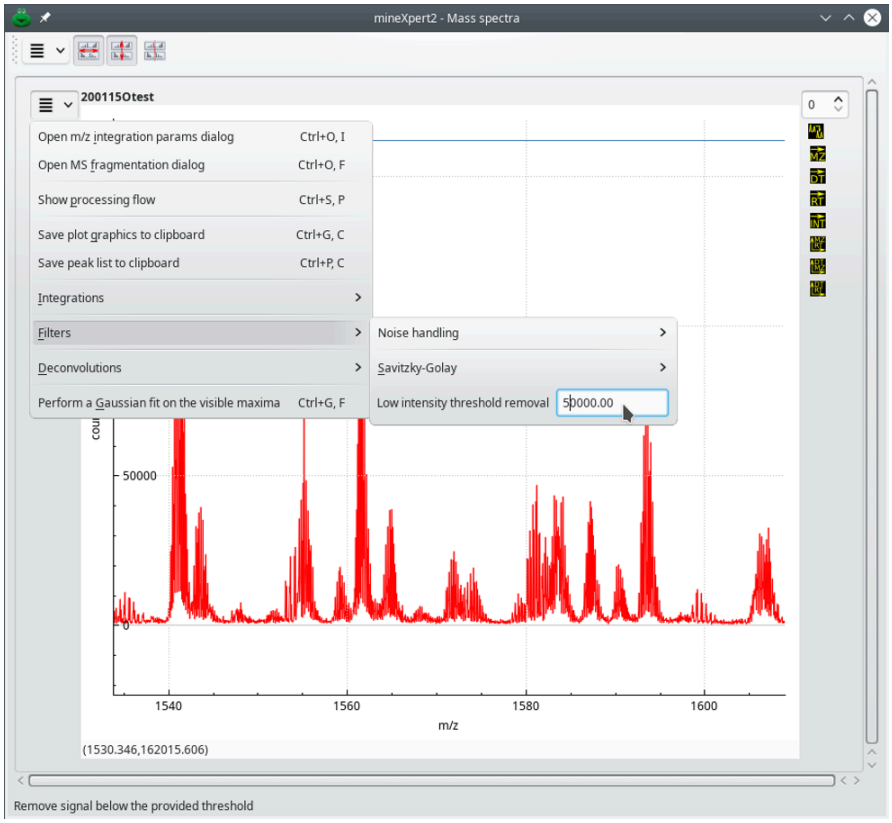


Figure 56: Example of low intensity signal removal (configuration)

The menu item shows the Low intensity threshold removal menu item. The line edit widget allows one to enter the threshold below which signal is to be removed from the trace. Once the proper value has been entered, the user clicks the menu text. The program runs the filter and produces a new trace that is displayed along with the original trace.

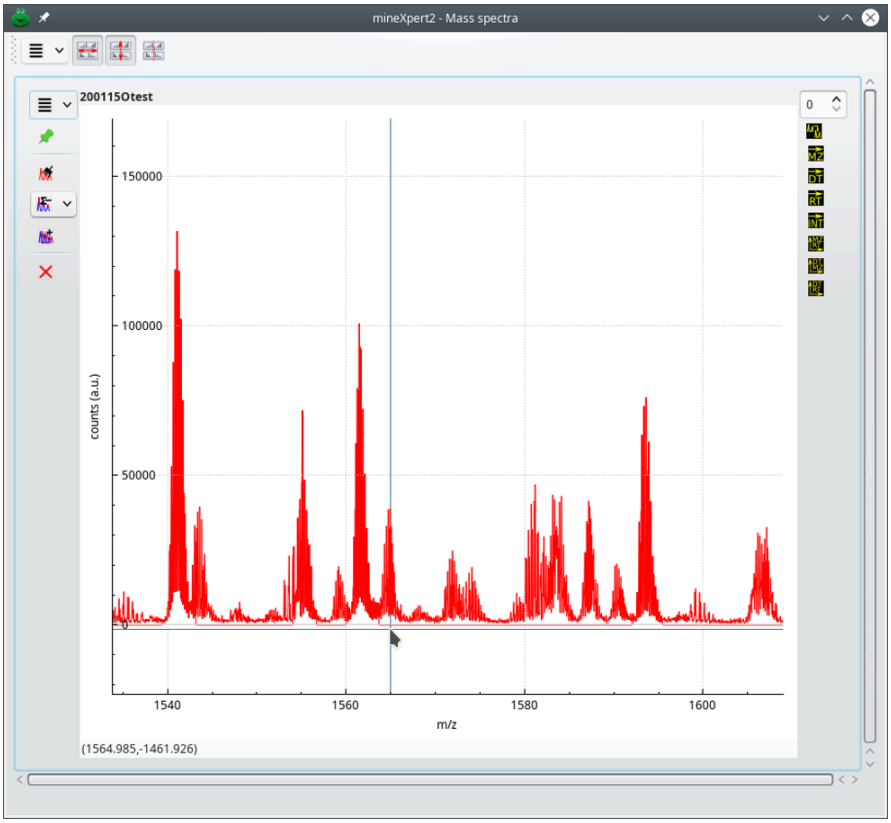


Figure 57: Example of low intensity signal removal (creation of a new trace)

The new trace is displayed in the same composite plot widget. The user needs to zoom in enough to find it. Here, it is clearly visible at the location of the cursor near the baseline. It appears as a straight line under the mass peaks that were filtered out. In order to better visualize the new trace, select it and duplicate it using the duplication icon button at the top of the functional button on the right hand side of the composite plot widget.

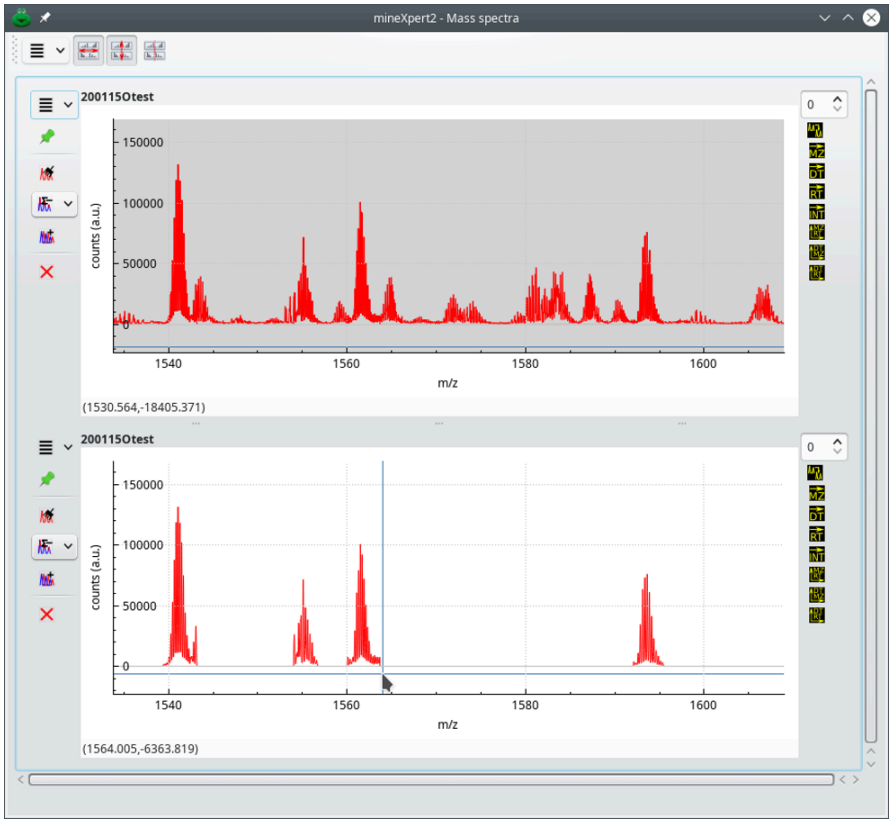


Figure 58: Example of low intensity signal removal (display of the new trace)

As is visible in the image, a great amount of signal present in the original mass spectrum (top plot) was removed from the trace (bottom plot). Note that the line connecting the isotopic cluster is faint where mass peaks were removed, but it is actually there. Note some remaining artifacts right after the very first isotopic cluster in the filtered trace. This is experimental work on progress.

5.6 Chained Integrations

The user, in the process of exploring the data, will inevitably chain integrations to pinpoint a specific feature of interest. For example, let's say that the user is exploring ion mobility mass spectrometry data.[10] After having loaded the data file, the TIC chromatogram is computed and displayed. From there, it is possible to perform a TIC chromatogram to an intensity = f(dt,*m/z*) colormap integration (see Figure 16 – The int = f(dt, *m/z*) colormap window.)

[10] Thanks to Dr Valérie Gabelica (IECB, Bordeaux, France) for permission to use her data set.

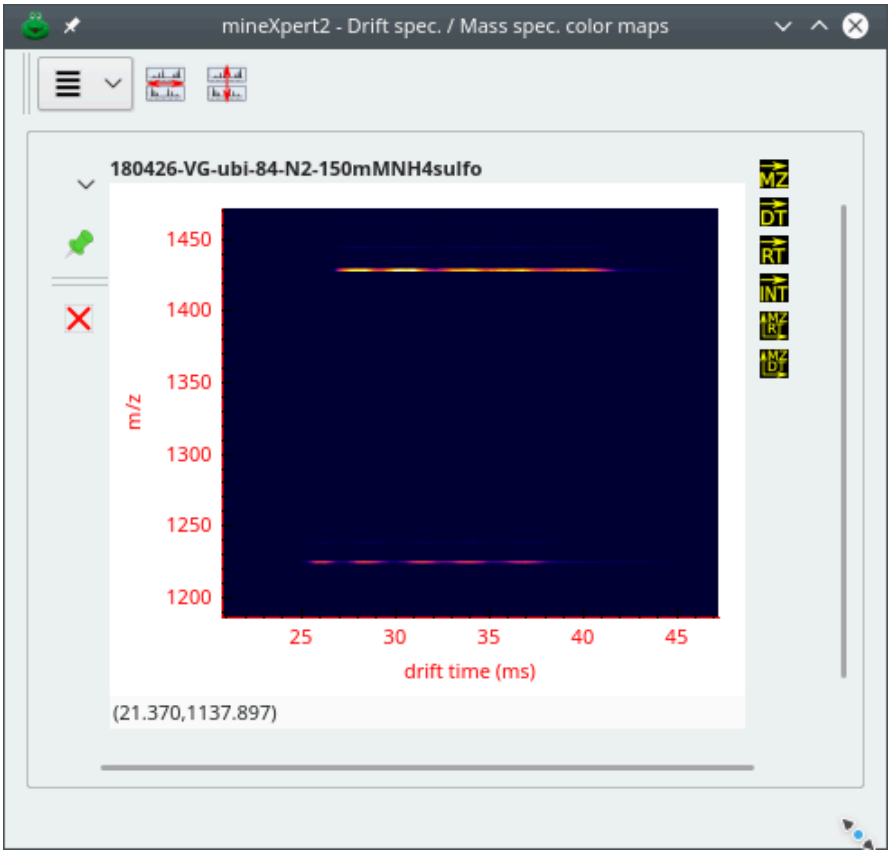


Figure 59: Example of chained integrations, colormap where data exploration may start

There starts the exploration. The user sees that there are a number of species having discrete drift times at the *m/z* ratio around 1220 (lower region of the colormap). They thus integrate to a single drift spectrum that horizontal lower region of the colormap. The obtained drift spectrum is shown at the left hand side of Figure 60 – Multiple chained integrations.

Because there are five drift peaks in the drift spectrum, the user performs as many individual mass data integrations to a mass spectrum, from left to right. The obtained mass spectra are all shown in the window next to the Drift spectra window of the same figure.



Figure 60: Multiple chained integrations

Most interestingly, the various drift regions are integrated to almost identical *m/z* values in their respective mass spectrum. In order to know when the various molecular species eluted in the chromatogram, the user performs for each mass spectrum an integration to a XIC chromatogram. The XIC chromatograms are shown on the window next to the Mass spectra window. Visibly, each molecular species was eluting from the chromatography column at discrete retention times (this was clearly not a *true* chromatography but instead an infusion during which instrument parameters were changed to modify the mobility properties of the ubiquitin molecule).

At this point, it is interesting to confirm that each XIC chromatogram contains exactly the molecular species that were responsible for the appearance of the purple "bands" on the int = f(dt, *m/z*) initial colormap. The different XIC chromatograms are thus integrated to a colormap and the results are shown below the initial colormap (less one colormap that did not fit in the window). Visibly, the extracted colormaps reconstitute the initial pattern visible at the top of the Drift spec / mass spec colormaps window.



Note

The dataset used for Section 5.6 – Chained Integrations was kind courtesy of Dr Valérie Gabelica and correspond to a work entitled *Optimizing Native Ion Mobility Q-TOF in Helium and Nitrogen for Very Fragile Noncovalent Structures* published in *JASMS* with DOI: 10.1007/s13361-018-2029-4.

5.7 Multi-widget Intensity Integrations

It happens pretty often, when doing quantitative mass spectra data analyses, that intensity integrations are desirable for a set of available mass spectra.

Typically, this would be the case when assessing the quantitative differences between a number of samples, the mass spectral data of which have been integrated to mass spectra.

To perform an intensity integration for multiple composite plot widgets, first pin down the widgets of interest (see here for an explanation about pinning down composite plot widgets), ensure that there is either only one trace in each one of them or the trace of interest is selected and perform the integration as usual in any one of the composite widets, *but* with the `Shift` keyboard key pressed all along. The integration results will be printed to the console window using the plot color, along with othe discriminating metadata.

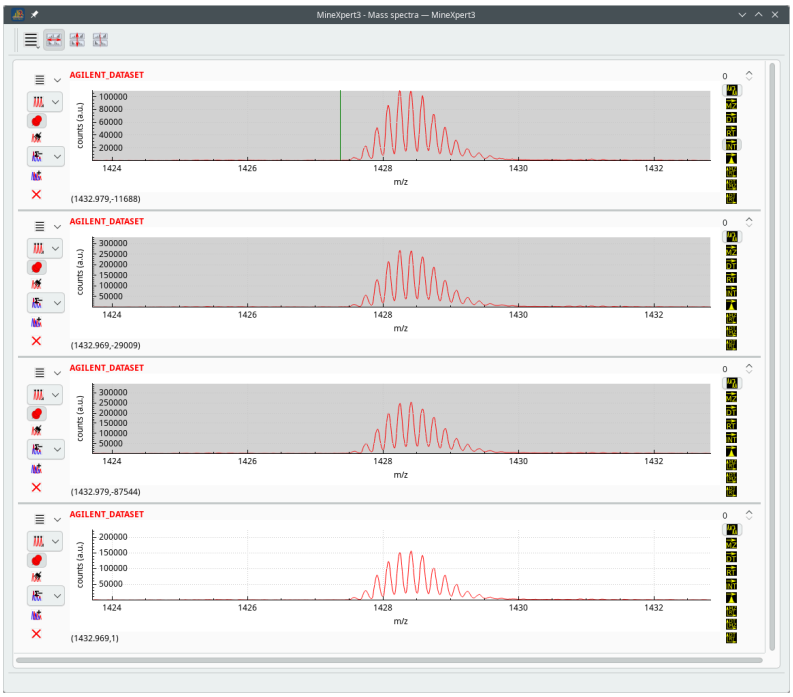


Figure 61: Performing simultaneous integrations to intenstity on multiple composite plot widgets

All the composite plot widgets are pinned-down and `Shift` is maintained pressed upont dragging for an integration to intensity.

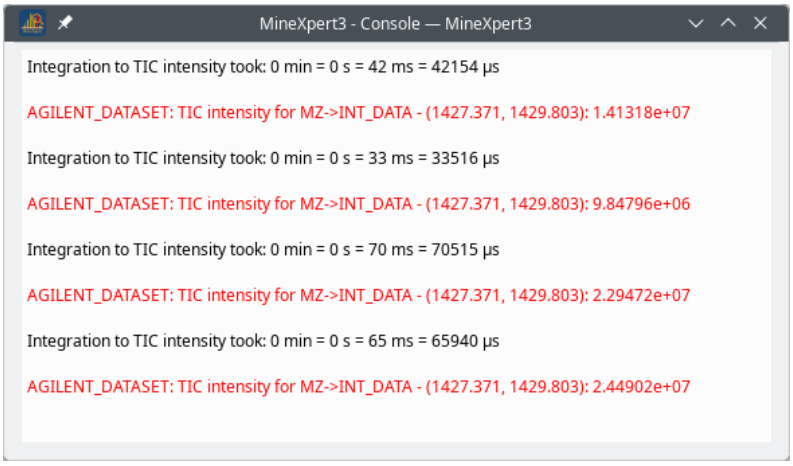


Figure 62: Results from an integration to intensity on multiple composite plot widgets

The console window details all the computed intensities. The integration range is indicated in parenthesis.



Note

In the multi-widget intensity integration, the plot range that is selected for integration by doing the `Shift` -click-drag movement is used for all the traces found in the pinned-down widgets, *whatever the currently shown plot* m/z *range*. It is thus advised, when performing multi-widget intensity integrations, to lock all the traces in the x-axis.

In the multi-widget intensity integration, it is possible to trigger the cumulative intensity integrations, as described at Section 5.8 – Cumulative Intensity Integrations.

5.8 Cumulative Intensity Integrations

It happens pretty often, when doing quantitative mass spectral data analyses, that cumulative intensity integrations are required. This is the case, for example, when a given analyte species that is present on a mass spectrum as ions of different charges needs to be quantitatively accounted for under a unique quantitation value. Another example is when oxidized and non-oxidized analytes need to be accounted for under a unique quantitation value. Finally, another example is when ananalyte is present on the mass spectrum as a free ion and as a sodium and/or potassium adduct; in this case the analyte quantitation needs to account for all two or three species.

MineXpert has a feature for this specific case that is activated using the composite plot widget main menu's **Integrations** menu item and *Sum up intensities* sub menu (Figure 63 – Starting cumulative intensity integrations).

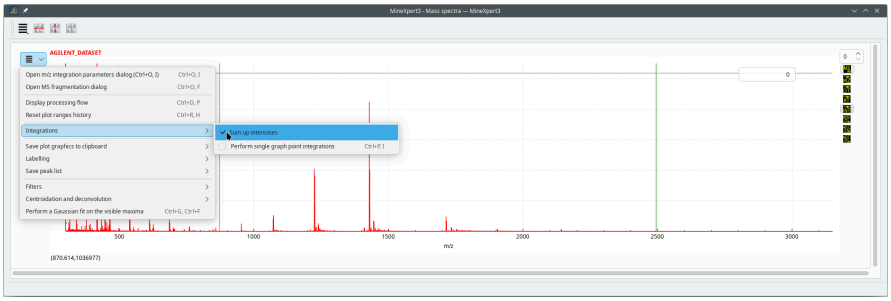


Figure 63: Starting cumulative intensity integrations

When the cumulative intensity integrations are initiated, an empty small line edit widget at the top right corner of the plot widget is displayed.

In Figure 64 – Performing multiple intensity integrations the user has cumulatively integrated the intensities of the same analyte mass spectral signal under three peaks charged 5+, 6+, and 7+.

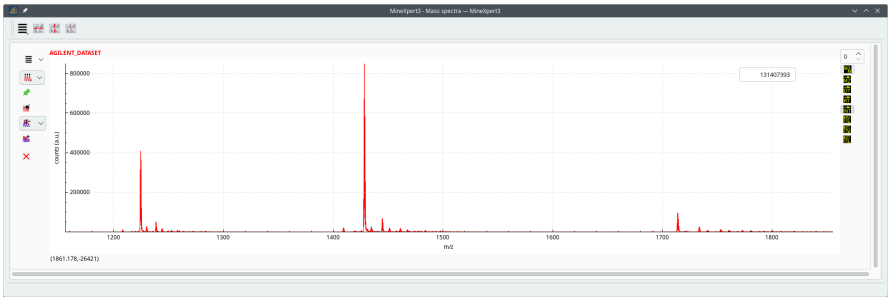


Figure 64: Performing multiple intensity integrations

When cumulative intensity integrations are performed, the newly calculated intensity values are added to the value that is displayed in the small line edit widget at the top right corner of the plot widget.

---

6 Centroid Extraction from Mass Spectra

> **Note**
>
> A general overview of the centroid extraction ⟶ deconvolution ⟶ analyte identification workflow is illustrated at Figure 85 – Workflow leading from a mass spectrum to analyte identity suggestions.

The low mass deconvolution algorithm can be tightly configured using a large set of parameter settings that are detailed below and that are accessible via the application preferences window shown in Figure 78 – Parameter set governing the low mass deconvolver and detailed below.

The centroidation of a mass spectrum is also called peak picking. This process is aimed at converting a profile-mode spectrum, that has its peaks properly shaped in Gaussian-like form, into a set of *m/z* values that recapitulate the initial peaks.

The configuration of the centroid extraction process is performed in the application preferences window, under the Centroidation section, as shown in Figure 65 – Configuration of the centroidation of mass spectra. There are two aspects in that configuration: the signal-to-noise estimator needs to be configured for the peak picker to be able to actually pick peaks. The peak picker needs also to be configured.

One aspect that is very important in peak picking is the determination of what is considered to be noise and what is considered to be actual signal. The signal-to-noise estimation is an optional task that is only carried out when the Signal to noise ratio is set to a value greater than *0*. The signal-to-noise estimator needs a number of parameters to be set up (in the Configuration of the signal to noise estimator group box widget). One important configuration bit is related to the determination of what intensity value is considered to be the maximum intensity value over the whole mass spectrum. There are multiple ways to define that maximum intensity value, as described in Figure 65 – Configuration of the centroidation of mass spectra.

- Manual setting: this is the simplest method to define that maximum intensity value, by having the user setting it up.
- Standard deviation factor: the software first makes a statistical analysis of the intensity values throughout the mass spectrum and any data point that has an intensity over the standard deviation compounded by this factor is discarded.

- Max percentile: any data point that has an intensity over the intensity of the data point representing the top 5 percent of the intensity value space of the mass spectrum is discarded.

Other configuration bits are the Rolling window size (m/z) which specifies the *m/z* range considered when computing the local signal-to-noise ratio. The rolling window is moved data point by data point over the whole mass spectrum. Each mass spectrum data point has thus a corresponding signal-to-noise ratio value. The Intensity histogram bins configuration bit is the number of bins in the histogram of the intensities in the mass spectrum. This parameter defines the granularity with which the signal-to-noise ratio is computed for each mass spectrum data point. The Mininum required elements configuration bit is the number of data points that are required to be found inside the rolling window.

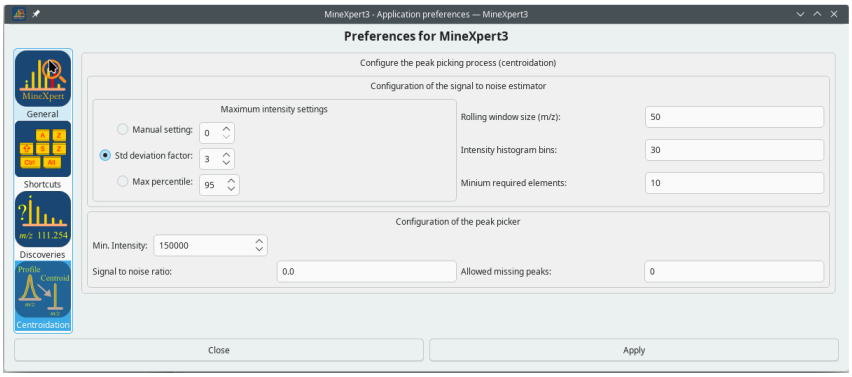


Figure 65: Configuration of the centroidation of mass spectra

---

The peak picker needs the following bits of configuration. The Min. intensity configures the signal intensity below which the centroids are not extracted. This is useful for highly noisy traces or when one knows that the signal of interest is above a determinat value. The Signal to noise ratio value determines the ratio below which the peak centroid is not extracted. Setting this value to 0.0 will disable the signal to noise process step. Finally, one last bit of configuration for the centroid extraction is the Allowed missing peaks value that defines the number of *m/z* values that can be found to be absent from the rolling window mentioned above. This parameter provides some leeway when dealing with sparse mass spectra.

The menu to start the centroid extraction is described in Figure 66 – Menu to be used to extract centroids from a trace. The figure show the mass spectrum for which centroidation is asked. When the centroidation is actually performed, the centroids are plotted in the form of a bar graph, colored in blue (see Figure 67 – The successfully extracted centroids are displayed in the form of a bar plot).

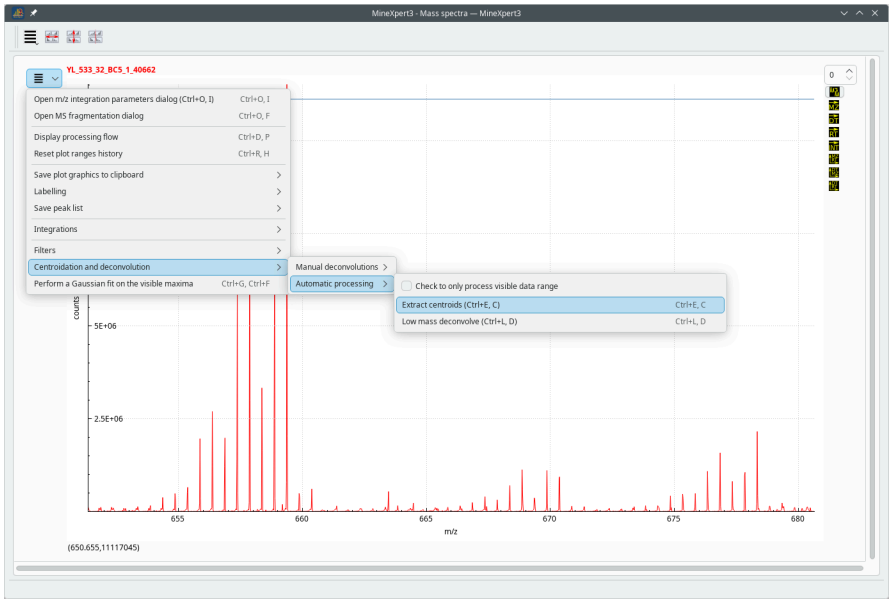


Figure 66: Menu to be used to extract centroids from a trace

Centroid extraction from a profile mode mass spectrum.

When the centroid extraction process is terminated, the centroids are plotted as a bar graph. These centroid data points can be exported to a file or a clipboard exactly the same way as for the normal trace graphs (see Figure 67 – The successfully extracted centroids are displayed in the form of a bar plot).

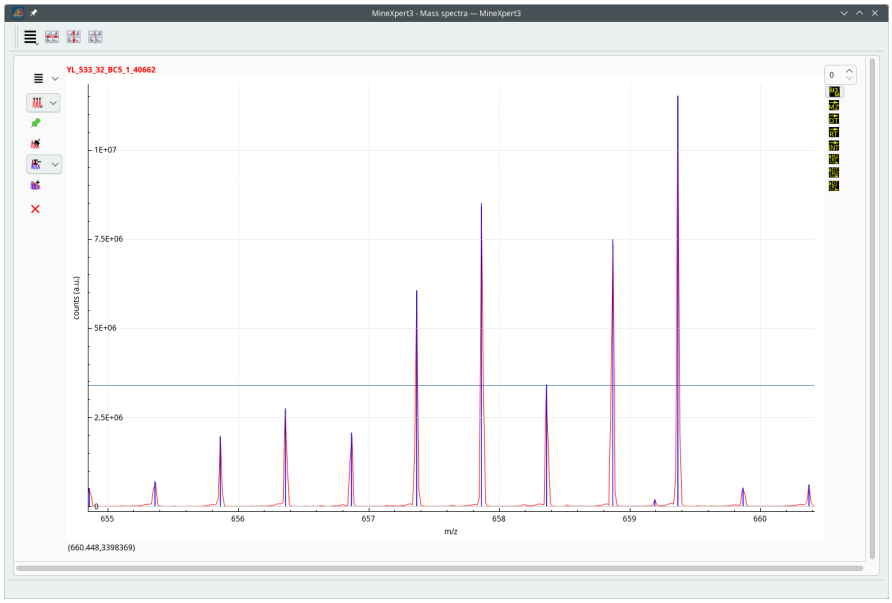


Figure 67: The successfully extracted centroids are displayed in the form of a bar plot

The centroids are plotted as a bar graph that can be exported to the clipboard or to a file exactly like for any trace mass spectrum.

← Centroid Extraction from Mass Spectra

Mass Spectral Deconvolutions →

> **Note**
>
> Upon successful extraction of centroids, these are output to the console window.

> **Warning**
>
> The centroids bars plot is not of the same kind as the plot from which the centroids were extracted. Please, make sure to read section Section 9.2 – Recording Processing Steps Detached from the Initial Mass Spectrometry Dataset.

7 Mass Spectral Deconvolutions

When analysing a mass spectrum, there are two available deconvolution modes:

- Manual deconvolutions:
  - Isotopic cluster-based deconvolutions (Section 7.2 – Deconvolutions based on isotopic cluster peaks);
  - Charge state envelope-based deconvolutions (Section 7.1 – Deconvolution based on charge state envelope inter-peak distance);
- Automated deconvolutions (?? Unknown label: <sec_automated-isotopic-cluster-based-deconvolutions> ??)

Deconvolutions are performed in order to get back to the M_r mass of the analyte while reading *m/z* values. The deconvolution processes below first determine the charge of the ion beneath the mass spectral data (either charge envelope states or isotopic cluster). Then, if the user has pointed a *m/z* value precisely, the process goes through to the molecular mass of the analyte (the ionization agent is always considered to be the proton entity). In the following sections, all the available deconvolution processes are described in detail.

Before delving into the deconvolutions, it is necessary to present two menu options that are found in the plot widgets contained in the Mass spectra window: the menu items under the Centroidation and deconvolutions menu (Figure 68 – Mass spectrum plot widget-specific deconvolution menu).

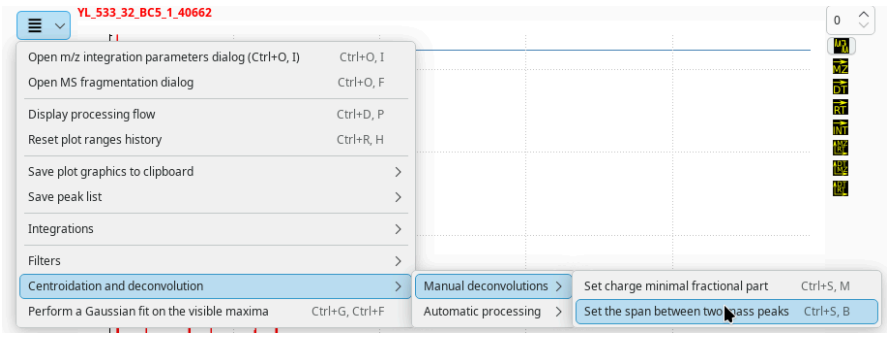


Figure 68: Mass spectrum plot widget-specific deconvolution menu

The two menu items are needed to configure the mouse-based deconvolution of mass spectra.

These two menus allow one to set parameters for the manual deconvolutions (see text for details).

7.1 Deconvolution based on charge state envelope inter-peak distance

In this kind of deconvolution, at the present time, the software assumes that the ionization agent is the proton and that the ionization is positive.

The deconvolution is based on the determination of the distance between two peaks – consecutive or not– of a given charge state envelope. When the user ⌥-click-drags the cursor from one peak to another, the program tries to calculate if the distance between two peaks matches one or more charge difference(s). If so, it computes the molecular (M_r ^{12}C-relative molecular mass) mass of the analyte.

> **The mouse drag movement is significant**
>
> The M_r value that is computed is for the analyte below the mass peak at which the mouse drag moment operation *started*.

Top panel of Figure 69 – Charge state-based mass deconvolution shows the charge state envelope-based deconvolution process for a protein of M_r#asymp8566 Da. In the top panel, the deconvolution has involved two consecutive peaks (*span* is 1). The mouse drag movement occurred from left to right. Thus, the *m/z* value chosen for the computation is that under the left peak (start point of the movement). The status line at the bottom of the plot indicates the selection range, the delta movement on the x-axis, the computed charge, the M_r value chosen for the calculation (the M_r value at the *start* of drag movement) and finally the calculated M_r value.

This “two consecutive mass peaks” method is the default method. However, it might happen that no clearly visible mass peak is available around one nice peak that might be chosen as the start of the mouse drag operation. In this case it is possible to define a different *span* between two peaks elected for the deconvolution (see Figure 68 – Mass spectrum plot widget-specific deconvolution menu). In the figure, that span has been set to 2, which means that the mouse drag movement encompasses three mass peaks: the movement start peak, one peak in the middle and finally the movement stop peak (the span is thus of two intervals between the extreme peaks).

The bottom panel of the figure now displays the same M_r value for the protein even if the span is now of two intervals.



Figure 69: Charge state-based mass deconvolution

Deconvolution approach using two peaks belonging to the same charge state envelope. The top deconvolution involves two consecutive mass peaks (peak span value is 1). The bottom deconvolution involves two non-consecutive peaks (peak span value is 2). The M_r value, expectedly, did not change whatever the configured span.

Note

The charge calculation, which is at the heart of the deconvolution, almost never produces an integer value with no fractional part (say, charge z=15.0) because it is almost impossible to drag the mouse cursor the exact number of pixels that would match a m/z range leading to such an integral charge value. Almost always, the charge that is calculated looks like 14.995 or 15.001, for example. This is due to the fact that the mouse moves at discrete positions on the screen and these positions might be more or less far apart, depending on the mouse capabilities and on the current zoom factor over the mass spectrum region of interest. It is advised to zoom-in as much as possible over the peaks at hand so as to minimize the difficulties above. It may happen, however, that even zoomed-in peaks are not sufficiently distant to allow a charge calculation. In this case, reduce the stringency over the fractional part that is allowed in the charge (see menu item Set charge minimal fractional part at Figure

← Mass Spectral Deconvolutions

Analyte identifications →

68 – Mass spectrum plot widget-specific deconvolution menu). By default, the stringency is set at 0.99, that is, any calculated value that has a fractional part either superior or equal to 0.99 or inferior or equal to 0.01 would lead to a successful round-up/round-down to the nearest integer value. Outside of the [0.01-0.99] interval, no charge calculation is performed and thus no deconvolution is performed. When the stringency is too high, reducing it will allow the deconvolution to be carried-over. General experience is that setting that value to 0.997 is fine for most situations and provides very reliable results.

7.2 Deconvolutions based on isotopic cluster peaks

There are two different (albeit similar) deconvolution modes that are based on the isotopic cluster representing the isotopologues of a given analyte:

- *The two-point mouse drag way:*: in this kind of deconvolution, the user ⌥-click-drags the cursor between the first two peaks (when possible) of the isotopic cluster. The charge state of the ion is the inverse of the distance between the two consecutive peaks (that is, the *m/z* delta value; see Section 7.2.1 – Two-point charge determination with Mr calculation).
- *The multi-point mouse drag way:*: in this kind of deconvolution, the user ⌥-click-drags the cursor starting at any one of the isotopic cluster's peaks up to any location in this isotopic cluster. The charge state of the ion is determined on the basis of the statistical analysis of the extracted centroids (which happens behind the curtains; see Section 7.2.2 – Multi-point statistical analysis-based charge determination).

7.2.1 Two-point charge determination with Mr calculation

This method is the most immediate one and is reliable when the user can visually pin-point the correct centroid position for two profiled isotopic cluster peaks. There is no centroid extraction in this calculation mode. The user has to drag the mouse from the first perceived centroid position to the next perceived centroid position. When the program encounters that the distance between the mouse start and current drag points is a multiple of the $1/z$ ratio (1 being a rounded value of the mass delta expected for a switch from a ^{12}C to a ^{13}C

isotope), it prints the z value (see Figure 70 – Isotopic cluster-based mass deconvolution for a description of the process).

> **Note**
>
> If the user has started the mouse drag movement right at the proper centroid position of the first isotopologue (non-labelled analytes) or of the last isotopologue (fully labelled analytes), then the molecular mass (M_r) can be computed based on the determined z charge. These two values are then set in memory and might be used when recording discoveries (see Section 10 – Recording data exploration discoveries).

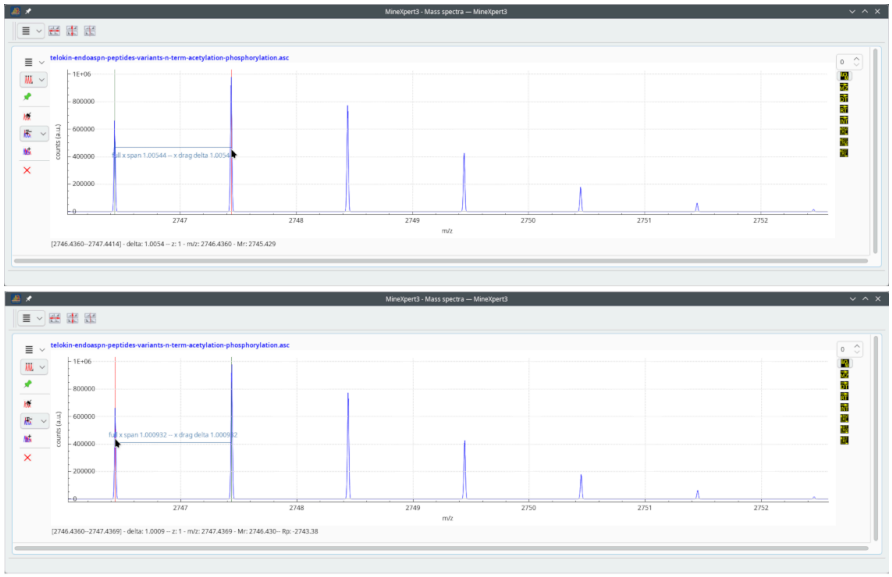


Figure 70: Isotopic cluster-based mass deconvolution

Top panel: deconvolution involving a left-to-right mouse drag movement. Bottom panel: deconvolution involving a right-to-left mouse drag movement. The calculated M_r values differ.

> **The mouse drag movement start position *is* significant**
>
> The 🖱-click-dragging direction (left⟶right or right⟶left) determines the final M_r that is computed because that value is calculated for the peak under the mouse when the mouse drag movement is initiated. This is visible in the two panels of Figure 70 – Isotopic cluster-based mass deconvolution, where the top panel shows the M_r computed for the left peak and the bottom panel shows the M_r computed for the right peak. Since the ion is monocharged, the difference is 1 Da.

This is a significant departure from the previous versions, where the postulate was that the single real peak of interest in an isotopic cluster was the left-most *monoisotopic* peak. Since this software has been used by scientists in research projects using almost 100 % labelled bacteria (with ^{13}C and ^{15}N), that concept has become moot. Indeed, analytes from these bacteria have their monoisotopic peak at the far right end of the isotopic cluster.

The new behaviour allows scientists to compute the M_r value of the peak of interest in an isotopic cluster, be that for a non-labelled or for a labelled analyte. See the following articles as examples of heavy isotope almost full labelling of bacteria.

Heavy isotope labeling and mass spectrometry reveal unexpected remodeling of bacterial cell wall expansion in response to drugs. Atze H, Liang Y, Hugonnet JE, Gutierrez A, Rusconi F, Arthur M. *Elife*, 2022, doi: 10.7554/eLife.72863, PMID: 35678393.

Peptidoglycan-tethered and free forms of the Braun lipoprotein are in dynamic equilibrium in Escherichia coli. Liang Y, Hugonnet JE, Rusconi F, Arthur M. *Elife*, 2024, doi: 10.7554/eLife.91598, PMID:39360705.

(p)ppGpp modifies RNAP function to confer β-lactam resistance in a peptidoglycan-independent manner. Voedts H, Anoyatis-Pelé C, Langella O, Rusconi F, Hugonnet JE, Arthur M. *Nat Microbiol*, 2024, PMID:38443580.

7.2.2 Multi-point statistical analysis-based charge determination

In this way of doing the determination of the charge of the analyte beneath an isotopic cluster of interest, the first step is to zoom in sufficiently to have a detailed view of the isotopic cluster at hand (see Figure 71 – Zoomed-in isotopic cluster for charge determination).

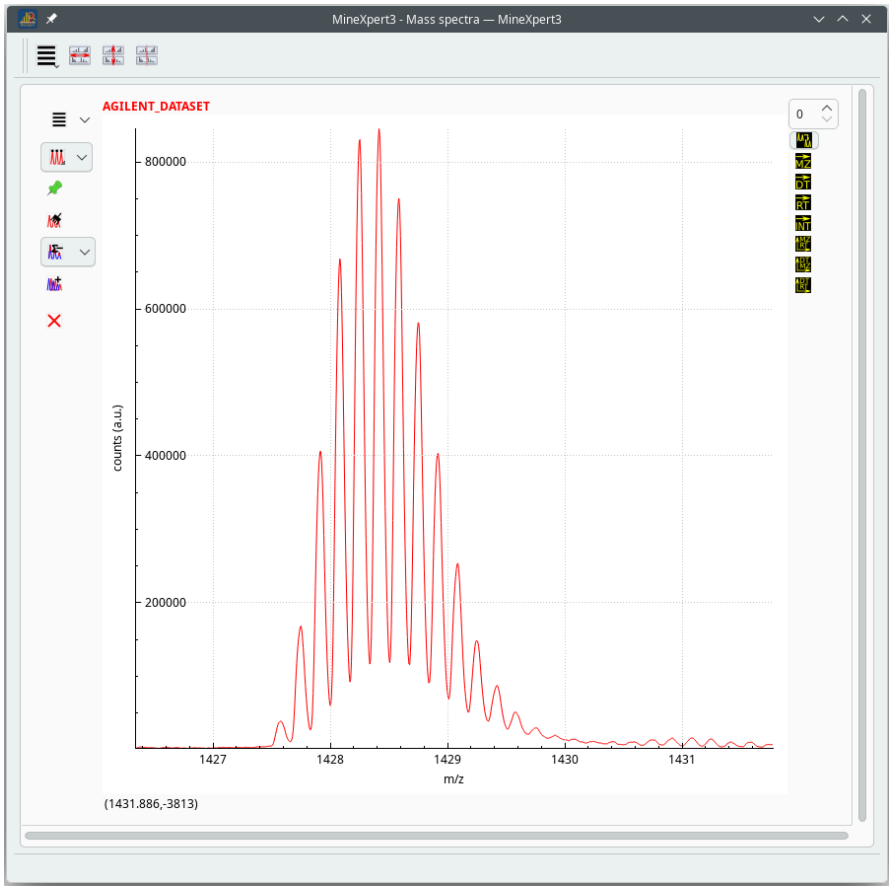


Figure 71: Zoomed-in isotopic cluster for charge determination

The starting point of the isotopic cluster-based multi-point charge determination is to zoom in the isotopic cluster at hand.

To trigger the automatic extraction of the centroids and the statistical analysis of the *m/z* spacings between these centroids, the user presses the C keyboard key, which displays the menu in Figure 72 – Triggering the isotopic cluster-based charge determination process.

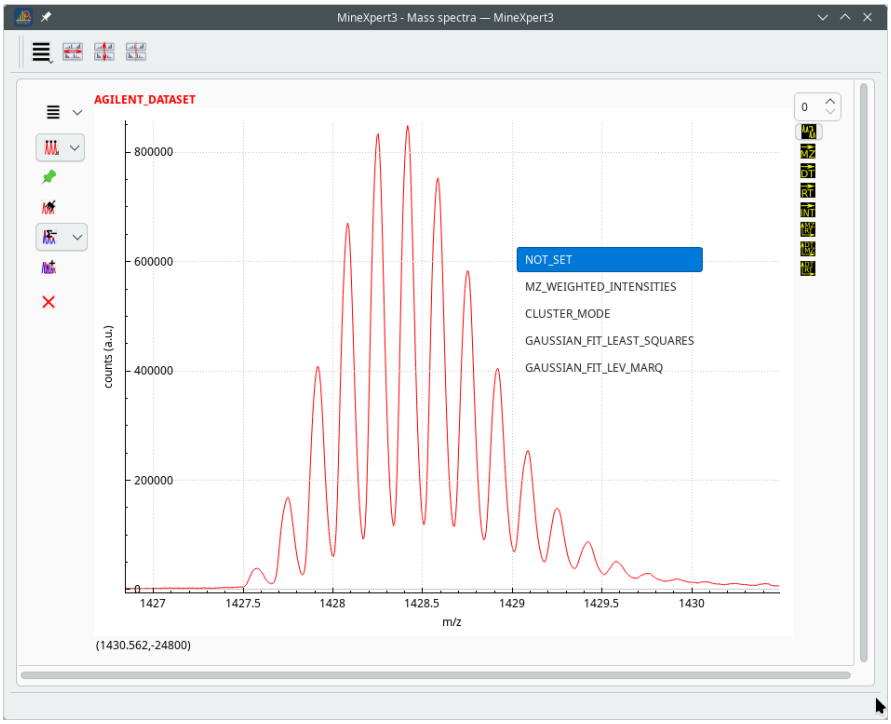


Figure 72: Triggering the isotopic cluster-based charge determination process

When the user presses the C keyboard key, the menu visible on the right hand side of the plot widget is displayed (or hidden, the key acts as a toggle). Depending on the menu item that is selected, various actions will be carried out when the user left-mouse-button-click-drags the cursor over the isotopic cluster.

Upon 🖱-click-dragging the cursor over the region of interest, the centroids are extracted and statistics about the extracted centroids are displayed in a text details widget. The process is shown in Figure 73 – Isotopic cluster-based charge determination, where three centroids could be extracted. As shown on the figure, the widget that appears provides some useful data about the isotopic cluster's centroids being extracted. There are three centroids currently extracted and initial statistics show that the putative charge of 6 may be considered reliable, although three centroids are not considered enough to actually compute an R^2 value (at least four extracted centroids are needed for this).

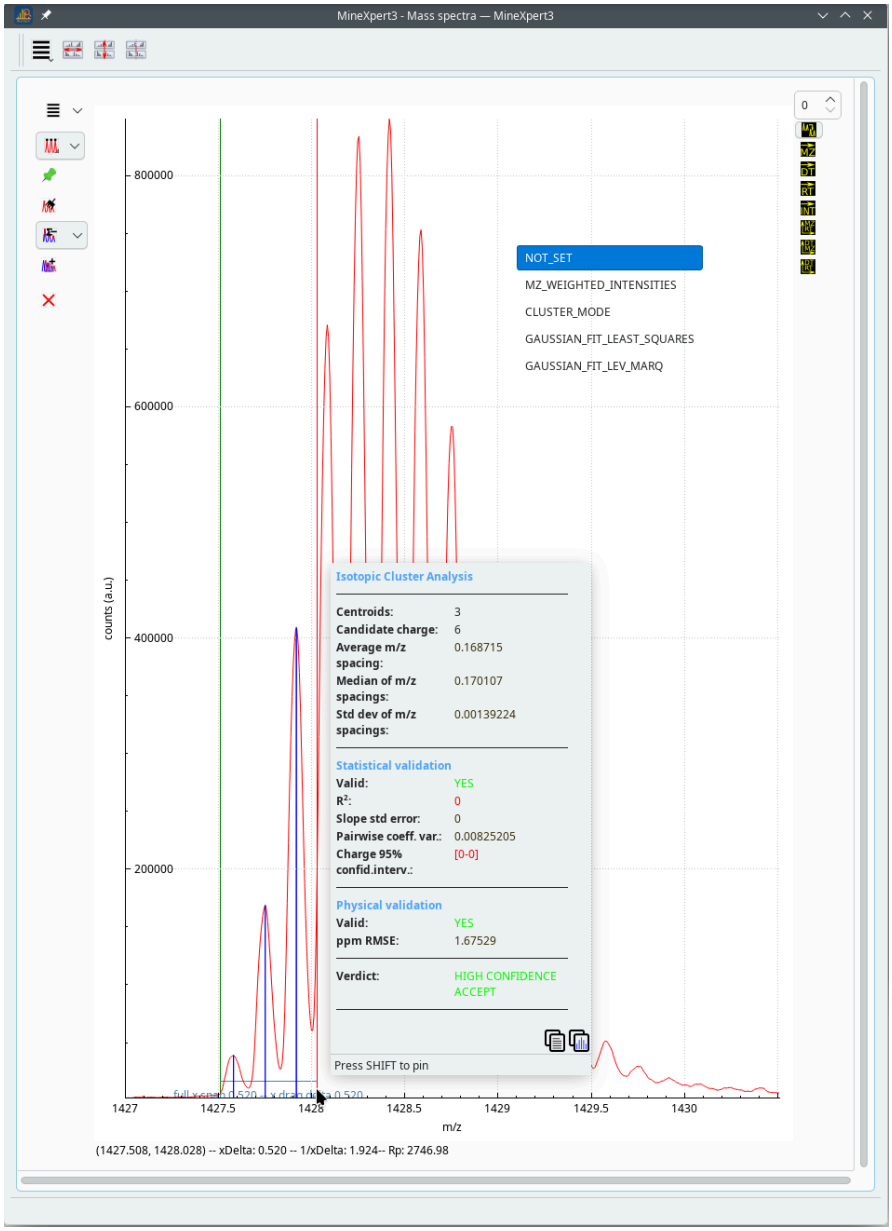


Figure 73: Isotopic cluster-based charge determination

When the statistics computed on the extracted centroids are not reliable, the verdict shows it. Here, the statistical validation could not be performed right because only three (3) centroids are available since the user left-mouse-button-click-dragged the mouse over too small a region. However, the distance between these three centroids is so consistent that the error between the observed distances and the expected ones is very small (1.5 ppm), leading the software to consider that the data are *physically* valid.

Upon continuing to extend the "selected" isotopic cluster region, the statistical analysis of the extracted centroids are updated, as shown in Figure 74 – Isotopic cluster-based charge determination.

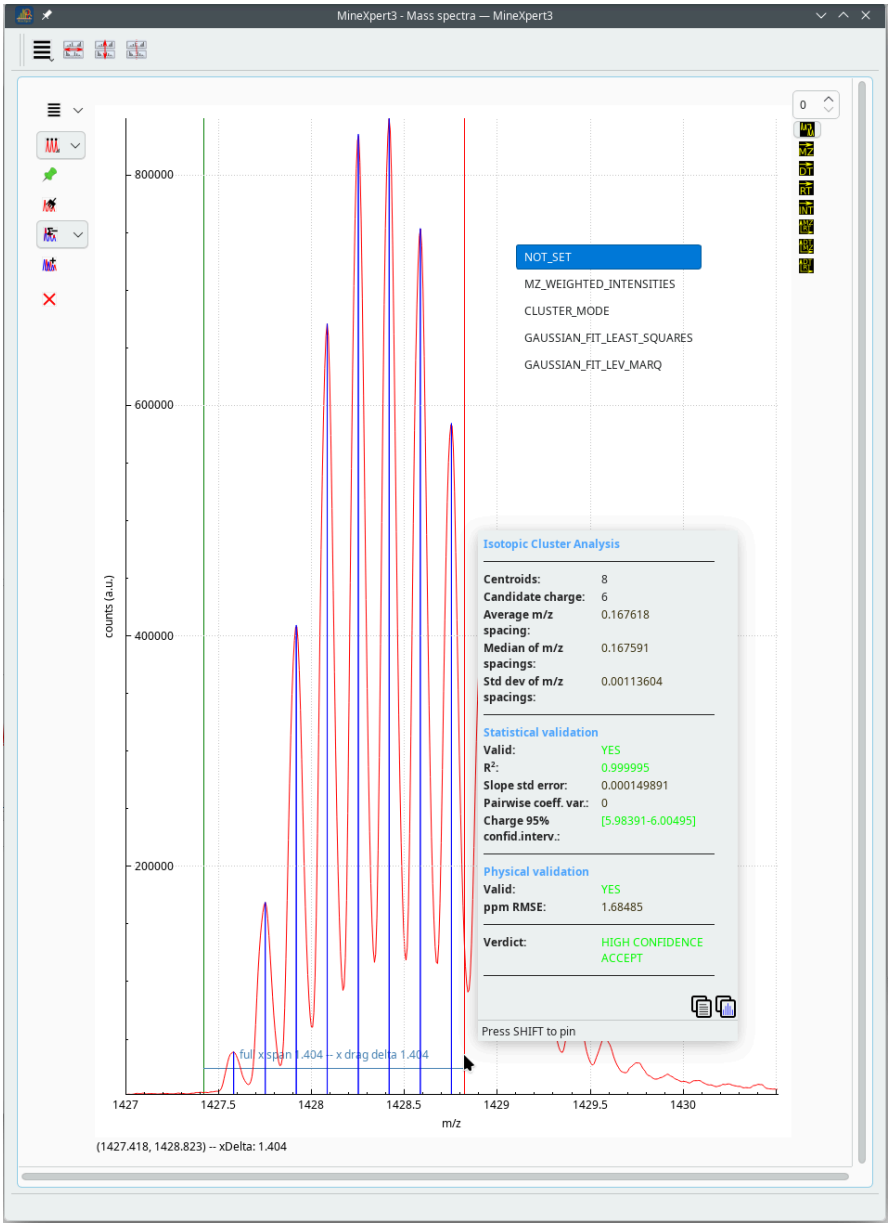


Figure 74: Isotopic cluster-based charge determination

When the statistics computed on the extracted centroids are reliable, the verdict shows it. In this situation, the user left-mouse-button-click-dragged the mouse over a sufficiently large isotopic cluster region to extract a sufficient number of centroids for the statistics on the spacings between them to provide a reliable charge determination.

Exporting the isotopic cluster-based charge determination statistics and/or the extracted centroids to the clipboard

When the user swiftly releases the ⇳ and immediately puts the cursor on the statistics-displaying widget, the widget stays in place as long as the cursor hovers over it. This allows the user to scrutinize the statistical data shown on the widget.

← *Mass Spectral Deconvolutions*

Analyte identifications →

Alternatively, the user may press the `Shift` keyboard key to force the text details widget to remain visible even when the cursor is not in its area.

Once the text widget details textual data have been scrutinized, if they are deemed of interest these might be copied to the clipboard by clicking the button at the right bottom corner of the widget. If the centroids are of interest, these may be copied by clicking onto the other button.

Note

The centroids that are extracted in this process are extracted according to the same process as that described in Section 6 – Centroid Extraction from Mass Spectra.

The centroids bars plot is not the same kind of plot as the plots obtained through conventional mass data integrations. Please, see Section 9 – The Processing Flow in a Data Exploration Session and in particular Section 9.2 – Recording Processing Steps Detached from the Initial Mass Spectrometry Dataset.

See Section 4.8.4 – General Plot Management in the Composite Plot Widgets for details about the handling of plots.

The configuration of the level of stringency for the statistics to be considered reliable for the charge determination is performed in the application preferences window, on the specific section entitled Charge determination, as shown in Figure 75 – Isotopic cluster-based charge determination.

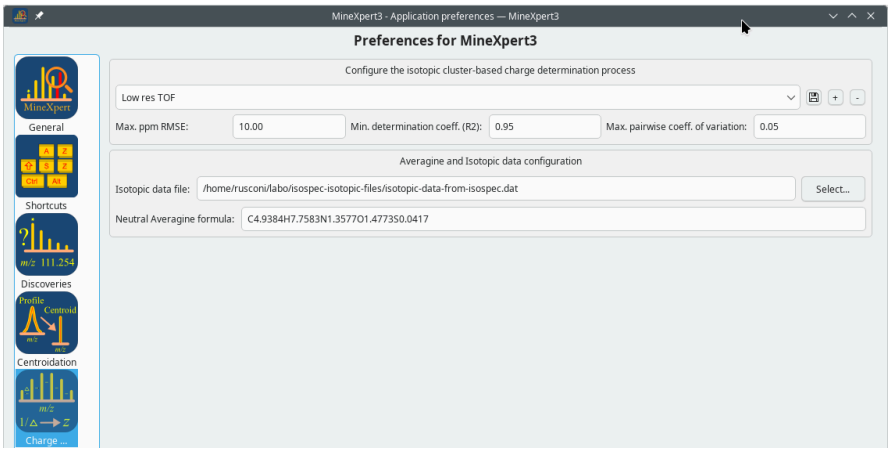


Figure 75: Isotopic cluster-based charge determination

Depending on the instrument used to acquire mass spectrometric data, the statistical analysis outcome stringency may be relaxed or tightened.

The statistical criteria involved in the assessment of the reliability of the determination of the charge are described below.

> **Note**
>
> The tool tips that display when the user hovers over the controls of the interface contain very detailed explanations of the statistical meaning of the variables set in this parameter section.

- `Max ppm RMSE`: This is a physical validation test on the charge value determined by the analysis of the isotopic cluster. With the centroid peaks' *m/z* value, determine the median of the *m/z* spacings between the various centroids' m/z value. The inverse of that median value is the putative charge. Next, compute the set of theoretically expected isotopologue centroids by anchoring the computation on the very first centroid extracted from the isotopic cluster (for $z = 1$, the delta *m/z* expected between two contiguous centroids is 1.003355 Th). Then compute the actual error between the observed centroids' *m/z* values and the corresponding calculated values. Turn the error into a ppm error (ppmError). In a variable, store the sum of the ppmError*ppmError values (that is, ppmError to the square) for all the deltas. The Root Mean Square Error value is the square root of that sum divided by the number of centroids in the cluster. The RMSE value is thus an indication of the consistency of the inter-centroid spacings. Depending on the analyzer used, the RMSE tolerance should be set to different values. Typically, for an Orbitrap analyzer, that RMSE tolerance would be 3 ppm, while for a TOF analyzer, it might be as high as 10 ppm. This test is useful because it helps rule out a charge value determined by inter centroid spacings that seem conducive to that charge but that in reality are not accurate enough for the analyzer being used.

In the configuration shown, for the Orbitrap high resolution analyzer, a typical value that is accepted for this RMSE value is 3 ppm. For a low resolving power instrument (an old TOF analyzer, for example), that value might be set to roughly 10 ppm.

- Min. determination coeff (R#super([2])): This is a linear regression analysis correlation coefficient tolerance. The entire isotopic cluster is fit into a linear regression model, with, for the abscissa values the isotopologue indices and, for the ordinate values, the corresponding intensity. The model is the line equation, with the slope being the mean *m/z* spacing between the centroids. Then the coefficient of determination is computed to reflect the reality (or not) that the centroids are perfectly evenly spaced, as expected for an isotopic cluster. A value of R^2 less than 0.98 implies that the centroid *m/z* values are not rightly on the regression line, they draw a jagged line;
- Max. pairwise coeff. of variation: This pairwise statistics is performed when the number of centroids is less than 4. In this case it is not possible to perform a robust regression: the variance of the *m/z* spacings between the centroids is thus computed to extract the standard deviation. The coefficient of variation (CV) is the standard deviation divided by the mean of the spacings. A reliable isotopic cluster has a coefficient of variation of the *m/z* spacings < 0.02 (2%). If the CV is high, then that means that the *m/z* spacings are fluctuating, which might indicate poor centroidation.



Warning

The *z* value (the charge of the analyte as determined using this method), is never set to the analysis context and does not provide the value for the recording of the mining discoveries (see Section 10 – Recording data exploration discoveries). Likewise, the determined *z* value is not used in combination with the start point of the mouse drag movement to compute the M_r value. This is because the start point of the mouse drag movement is not necessarily at the precise centroid position of the first peak of the isotopic cluster (see Section 7.2.1 – Two-point charge determination with M_r calculation for the proper way of doing this deconvolution). Also, the determination of a compelling *z* value that is required for the calculation of a reliable M_r molecular mass cannot be achieved using the method described in this section because there are different outcomes of different reliabilities.

7.2.3 Multi-point statistical analysis-based charge determination and mass estimations

In the previous section we saw how to determine the charge of the ion beneath an isotopic cluster of interest. This is the case when the NOT_SET menu item is selected in Figure 72 – Triggering the isotopic cluster-based charge determination process.

Once the charge of the ion beneath the isotopic cluster has been reliably determined, it is possible to estimate masses for the ion and for the neutral analyte. The various ways to do this estimation work are configured in that same menu, as detailed below.

- MZ_WEIGHTED_INTENSITIES: this method relies on the extracted centroids to determine the average m/z value according to a baricentric measure of where the most intense peaks are located.
- CLUSTER_MODE: this method relies on the extracted centroids to determine the most intense centroid and use that to estimate the average m/z value.
- GAUSSIAN_FIT_LEAST_SQUARES: this method relies on the extracted centroids to fit them to a Gaussian curve using the least squares algorithm. The apex of the Gaussian fit curve is the average m/z value of the ion.
- GAUSSIAN_FIT_LEV_MARQ: this method relies on the extracted centroids to fit them to a Gaussian curve using the Levenberg-Marquardt algorithm. The apex of the Gaussian fit curve is the average m/z value of the ion.

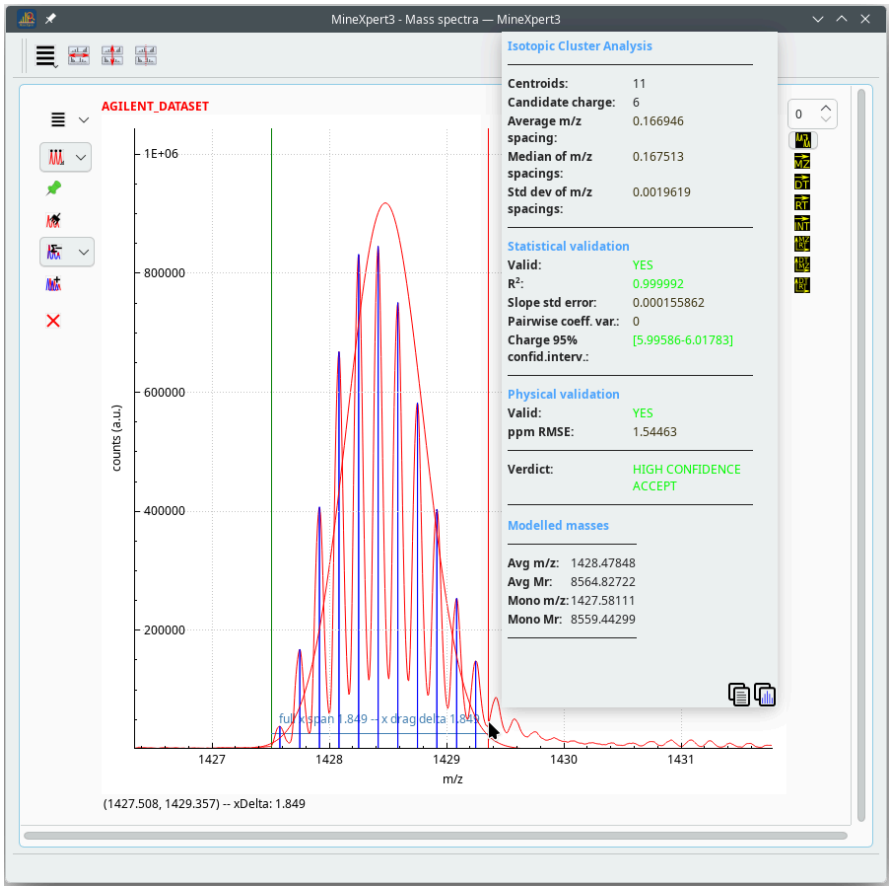


Figure 76: Gaussian fit of the isotopic cluster's centroids

The estimation of the masses of the ion beneath an isotopic cluster (average and monoisotopic) is performed through the fitting of the isotopic cluster's centroids into a Gaussian curve. The apex of the Gaussian curve is the average m/z value of the ion.

Once the average m/z value of the ion beneath the isotopic cluster has been determined, it is possible to estimate the monoisotopic mass of the analyte by using the *Averagine* model. The Averagine model is configured in Figure 75 – Isotopic cluster-based charge determination. For the computation to proceed correctly, it is necessary to provide both the path to the file that contains the isotopic data and the formula of *neutral* Averagine.

All the plots (be them graph plots, like for the Gaussian fit plot, or bars plots, like for the extracted centroids plots) are managed as described in Section 4.8.4 – General Plot Management in the Composite Plot Widgets.

Note

The Gaussian plot and the centroids bars plot are not the same kind of plot as the plots obtained through conventional mass data integrations. Please, see Section 9 – The Processing Flow in a Data Exploration Session and in particular Section 9.2 – Recording

Processing Steps Detached from the Initial Mass Spectrometry Dataset.

See Section 4.8.4 – General Plot Management in the Composite Plot Widgets for details about the handling of plots.

Cautionary note about the ion charge determination and the mass estimations

The features described in the sections above are only reliable when the quality of the isotopic cluster is sufficient to extract centroids that faithfully describe the isotopologues below each isotopic cluster's peak.

7.3 Low Mass Automated Deconvolution Based on Fully Resolved Isotopic Clusters

Note

A general overview of the centroid extraction⟶deconvolution ⟶analyte identification workflow is illustrated at Figure 85 – Workflow leading from a mass spectrum to analyte identity suggestions.

The low mass automated deconvolution that uses the isotopic cluster peaks is designed to be reliable for *low-mass* analytes (typically below 5–7 kDa). Deconvolution of a mass spectrum is triggered by selecting the menu shown in Figure 77 – Low mass deconvolution of mass spectral data.

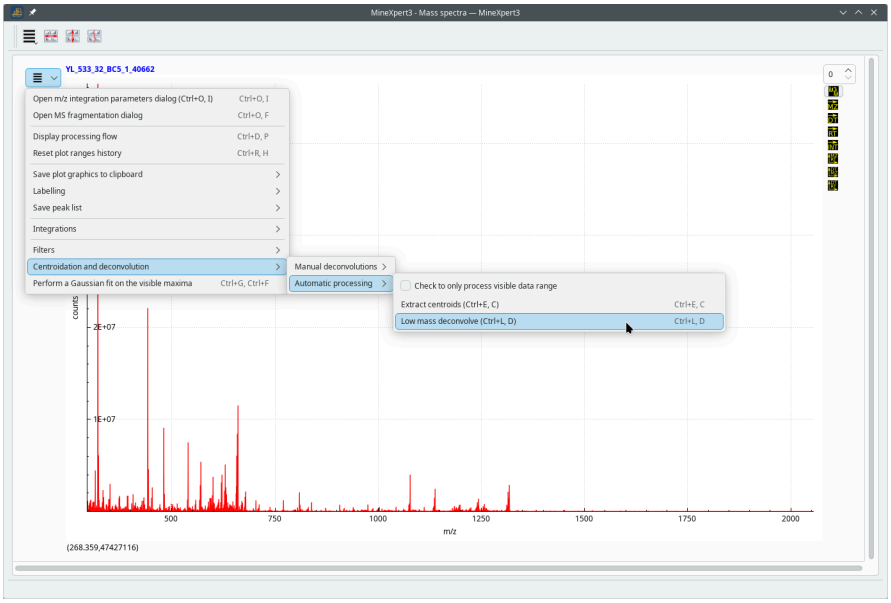


Figure 77: Low mass deconvolution of mass spectral data

The low mass deconvolution is reliable for analytes having masses contained in the 5–7 kDa range.

The process involves two separate steps: first, the centroids are extracted from the input mass spectrum (see Section 6 – Centroid Extraction from Mass Spectra); second, the extracted centroids are used for the deconvolution proper. The deconvolution is a highly complex process that would require extensive knowledge to understand. The main concepts are the following:

- Iterate in each *observed* centroid of the input centroided mass spectrum (from smallest (*m/z*, i) values to greatest) and, for each centroid, run a nested iteration loop where each charge in the range specified by the user is used to compute a neutral analyte mass (see below).
- The analyte's neutral mass is then used to construct a *theoretically expected* (that is, calculated) neutral averagine-based isotopic cluster. Each centroid of the isotopic is then taken in sequence and subjected to a matching process.
- The matching operation takes place by first converting back the theoretical neutral analyte cluster centroids to a charged analyte *m/z* value (same charge as that initially used to craft the neutral mass); second, the *m/z* value is searched for in the input mass spectrum centroids. When an input spectrum *m/z* value matches that of the theoretical isotopic cluster, a match is stored.
- When all the centroids of the theoretical isotopic cluster have been iterated into, the set of found matches is checked for reliability. The checks are performed according to the parameters set by the user and described below. If the checks are all positive, then the match is stored for the next step.

- At the end of the processing of all the input centroids, all the found matches are processed in order to consolidate them into a neutral mass. Each neutral mass is listed along with all the *m/z* values that were actually matched (these *m/z* values correspond to the different charge states of the same analyte and are called *supporting ions*).

The low mass deconvolution algorithm can be tightly configured using a large set of parameter settings that are detailed below and that are accessible via the application preferences window shown in Figure 78 – Parameter set governing the low mass deconvolver and detailed below.

The low mass deconvolver settings interface enables the user to store groups of settings for easy recalling along different user sessions. Each relevant widget has an associated explanatory tool tip.

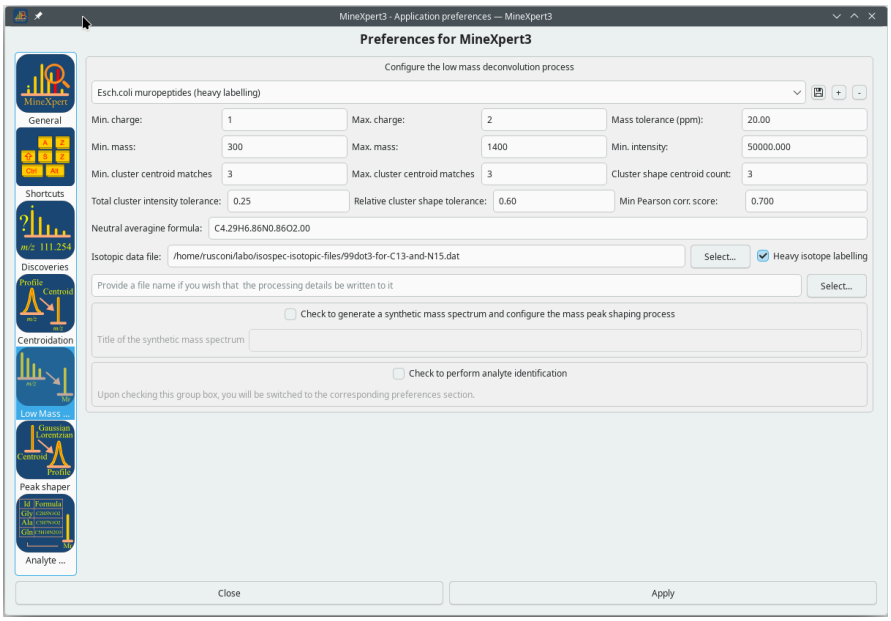


Figure 78: Parameter set governing the low mass deconvolver

Interface to the configuration of the low mass deconvolution process.

- Min. charge (Max. charge): the charge range that is explored.
- Mass tolerance (ppm): the tolerance with which all the mass matches are performed. A typical value is *20*. That values spans half of it on the left side of the mass value and half of it on the right side.
- Min. mass (Max. mass): the neutral mass range that is explored.
- Min. intensity: the intensity value below which input mass spectrum centroids are discarded.

- Min. cluster centroid matches: The minimum number of matching centroids between the actual input mass spectral centroids and the calculated theoretically expected centroid cluster.

- Max. cluster centroid matches: The maximum number of matching centroids between the actual input mass spectral centroids and the calculated theoretically expected centroid cluster.

- Cluster shape centroid count: Over how many isotopic cluster centroids the cluster shape quality test (see below) should be performed.

- Total cluster intensity tolerance: Once the matches between the centroids in the theoretical isotopic cluster and those in the input centroided mass spectrum have been performed, the reliability of these matches is checked like so: select the monoisotopic centroid and its contiguous centroid in both the theoretical isotopic cluster and in the matched centroided mass spectrum. Compute the ratio between the intensities of these centroids. That ratio is used to normalize the theoretical centroids intensities so they can be compared to the matched input mass spectrum centroid intensities. Compute the sum of the matched centroids intensities for all the matches (input mass spectrum vs theoretical). Compute the [input / theoretical] summed intensities ratio, which reflects the 'explained intensity'. The greater the ratio, the better the intensity correspondence between the theoretical and experimental isotopic clusters. That ratio value is required to be greater than the value set for this parameter for the matches to be considered reliable. Typical values: *0.5-0.7*.

- Relative cluster shape tolerance: Once the matches between the centroids in the theoretical isotopic cluster and those in the input centroided mass spectrum have been performed, the reliability of these matches is checked like so: get the monoisotopic (highest probability centroid, isotopologue 0) match. Get the isotopologue 1 match. For each one of these two matches compute the ratio of the intensities for the theoretical isotopic cluster data and for the input centroided mass spectrum (observed data) This yields the following intensity ratios: [theor_ratio = theor. intensity iso 1 / theor. intensity iso 0] and [obser_ratio -> obser. intensity iso 1 / obser. intensity iso 0]. Compute the ratio [abs(obser_ratio − theor_ratio) / theor_ratio] which amounts to a 'normalized difference'. If the ratio above is smaller than this parameter setting, then the matches are considered to be reliable. Typical values: *0.6-0.7*.

- Min Pearson corr. score: Once the matches between the centroids in the theoretical isotopic cluster and those in the input centroided mass spectrum have been performed, the reliability of these matches is checked like so: A Pearson correlation factor is computed to check that the observed and theoretical isotopic cluster centroids are correlated by

their intensity. If the obtained correlation factor is greater or equal to this setting, then the matches are considered to be reliable. Typical values: *0.7-0.90*.

- Neutral averagine formula: The averagine formula to be used to model the isotopic clusters starting from an analyte's neutral mass.
- Isotopic data file: The path name of the file holding the isotopic data. This is a file having the following format ([...]: removed data for clarity):

Isotopic data definition format

```
1,hydrogen,H,1,1.007825,1,0,0.999884,-0.000115,0
1,hydrogen,H,1,2.014101,2,1,0.0001157,-9.0644,0
2,helium,He,2,3.0160293,3,0,0.00000,-13.5206046 0
[...]
6,carbon,C,6,12.00000,12,0,0.9892119,-0.0108466,0
6,carbon,C,6,13.003354,13,1,0.0107,-4.529315,0
7,nitrogen,N,7,14.0030,-0.003648,0
7,nitrogen,N,7,15.00010,15,1,0.0036419,-5.615226,0
```



Note

It is possible to use an isotopic data file that lists heavy isotopes as the most probable isotopes for any chemical element (remember to check the Heavy isotope labelling check box). In this manual author's research, bacteria are fully labelled with the heavy stable isotopes of carbon and nitrogen, which produces peptidoglycan structural elements that display as inverted isotopic clusters (with respect to natural isotopic abundance analytes). The deconvolver works also in this case by only detecting the input mass spectrum isotopic cluster having the proper inverted shape, and reports the analytes with the diagnostic ions at monoisotopic *m/z* values corresponding to the highest value in the isotopic cluster that was correctly recapitulated by the averagine-based theoretical isotopic cluster. This means that whatever the isotopic abundance configuration, the reported ions are correct: their monoisotopic *m/z* value is reported.

- The last field may be used to indicate the path name to a file where all the processing details are stored. Perusing these data might be interesting when the deconvolution results do not correspond to expected results.

When the deconvolution process has ended, the results are shown in the console window.

If the user wants that a synthetic mass spectrum be created using the mass spectrometric data that could be used to obtain deconvolved features, checking the Check to generate a synthetic mass spectrum and configure the mass peak shaping process check box will switch the application preferences page to that entitled Peak shaper. The configuration of the peak shaper is described in ?? Unknown label: <sec_shaping-mass-peak-centroids> ??. The title of the synthetic mass spectrum can be specified in the corresponding line edit widget.

When the synthetic mass spectrum is ready, the user is provided with two options:

- Create a new MS run data set: This option creates a brand new mass spectrum.
- Select an existing data set: This option creates a brand new mass spectrum that is parented to the existing dataset.

The generated mass spectrum is plotted in a new plot widget, in a new color if it was not parented to the original mass spectrum, in the same color otherwise.

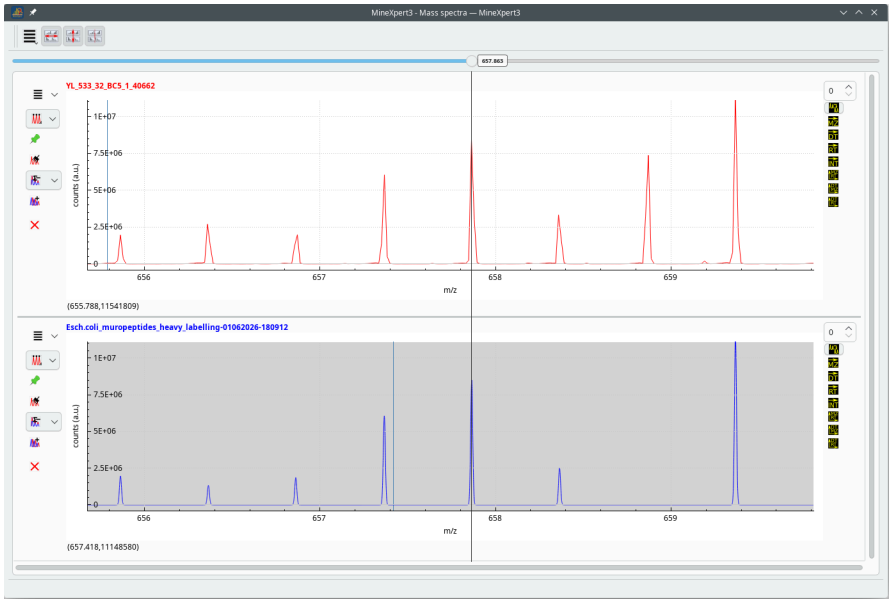


Figure 79: The synthetic mass spectrum (blue) mimicks the observed input mass spectrum only for successfully deconvoluted features

The synthetic mass spectrum is written to disk and loaded into the program. The user is informed to perform an integration to a mass spectrum.

Note

When performing an integration of the synthetic mass spectrum from the pseudo-TIC chromatogram, it is advised to check the *m/z*

integration parameters. Most often, setting these to “no bins” is the best option.

7.4 High Mass Automated Deconvolution Based on Fully Resolved Isotopic Clusters

The high mass deconvolution process is aimed at deconvolving mass spectra obtained on polymer analytes that have a large mass respective to the resolving power of the mass spectrometer. The mass peak obtained in such a situation for a large protein is thus a Gaussian-shaped peak the apex of which corresponds to the average m/z of the protein for a given charge state.

The deconvolution process is modelled around the UniDec project but does not use their code.

Please, do refer to this publication:

Michael T. Marty; Andrew J. Baldwin; Erik G. Marklund; Georg K. A. Hochberg; Justin L. P. Benesch; Carol V. Robinson *Bayesian Deconvolution of Mass and Ion Mobility Spectra: From Binary Interactions to Polydisperse Ensembles. Anal. Chem.* (2015) 87 (8): 4370–4376.

The parameters settings are complex and have been entirely regrouped in Figure 80 – High mass deconvolution parameter settings.

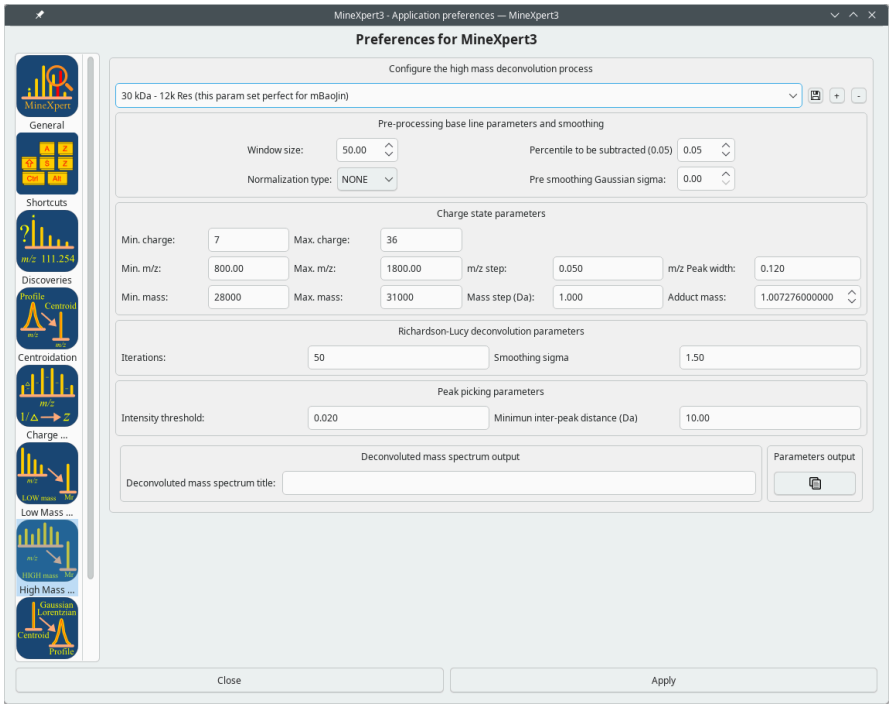


Figure 80: High mass deconvolution parameter settings

The high mass deconvolution process has a complex set of parameters to be adjusted.

The High Mass Deconv section of the application preferences window shows all the parameters that need to be set and adjusted for better results. We will review them one by one below.

First of all, it is possible to store across working sessions of set of parameters that might work particularly well for a given set of analytical data that is being explored. For thhis, first enter all the parameter values in the interface, then enter a meaningful name in the top combo box widget and click the + button. This creates a new set of parameters under that specific name. In order to have it stored on disk for accessing it later during another data exploration session, click the Apply button.

Whenever a value has been changed in that set of parameters, click the button next to the combo box widget that has a “diskette” icon. The set of parameters currently shown in the interface are recorded under the parameter set name visible in the combo box widget.

The parameters that need to be set for the high mass deconvolution process to work are described below:

- Pre-processing baseline parameters and smoothing:

  These parameters control the initial data conditioning applied to the raw input before the deconvolution engine begins.

  - Window size: Defines the width of the sliding window, in number of data points, used to calculate the local baseline across the spectrum. The baseline estimation

might be required to remove chemical/electronic noise from the raw input spectrum.

- Percentile: Specifies the percentile (from 0.0 to 1.0) of the intensity distribution within the baseline window that is used to define the baseline level. For example, a value of 0.05 uses the 5th percentile, effectively tracking the lower edge of the noise floor. Every thing above is reduced by the intensity value found for that percentile.
- Normalization type: The normalization method applied to the input intensities prior to processing. Options include NONE (no normalization), MAX (scales the maximum intensity to 1.0), or SUM (scales the total sum of all intensities to 1.0).
- Pre-smoothing Gaussian sigma: The standard deviation (in data points) of the Gaussian filter applied to the raw input data before binning. A value of 0.000 disables this pre-smoothing step.

- Charge state parameters:

  These parameters define the physical constraints and the mathematical grids used to map the m/z domain to the neutral mass domain.

  - Charge range: The minimum and maximum integer charge states (z) to consider when projecting the neutral mass grid into the *m/z* domain.
  - *m/z* range: The lower (min) and upper (max) *m/z* boundaries of the input spectrum to be processed. Data points outside this range are ignored.
  - *m/z* step: The bin size (in Th) for the regular *m/z* grid. The irregular, raw input data is interpolated or integrated onto this uniform grid prior to deconvolution.
  - *m/z* Peak width: The Full Width at Half Maximum (FWHM) of the peaks in the *m/z* domain, expressed in Th.
  - Mass range: The lower (min) and upper (max) neutral mass boundaries (in Da) for the output deconvolved spectrum. These are the likely masses that the analytes are expected to be deconvolved to.
  - Mass step (Da): The bin size (in Da) for the output mass grid. Smaller values yield higher mass resolution but will increase computation time and memory usage.
  - Adduct mass: The mass of the ionizing adduct (e.g., a proton at 1.007276 Da) used in the mass-to-charge conversion formula: $\frac{m}{z} = \frac{\text{Mr} + z \times \text{Adduct Mass}}{z}$.

- Richardson-Lucy deconvolution parameters:

  These parameters control the iterative mathematical engine that resolves the overlapping charge states into a clean neutral mass spectrum.

- Iterations: The total number of Richardson-Lucy iterative updates to perform. Higher values improve convergence and peak shape but increase processing time.
- Smoothing sigma: The standard deviation (in Da) of the Gaussian smoothing applied to the mass estimate during each Richardson-Lucy iteration. This acts as a regularization step to prevent noise amplification. Note: This value must be kept significantly smaller than the expected mass differences between species (e.g., < 16.0 Da when looking for oxidation ladders).

- Peak picking parameters:

  These parameters dictate how the software identifies and extracts individual peaks from the final deconvolved mass spectrum.

  - Intensity threshold: The minimum intensity threshold for peak detection, expressed as a fraction of the maximum intensity in the deconvolved spectrum (e.g., 0.020 means the peak must be at least 2% of the tallest peak).
  - Minimum inter-peak distance: The minimum allowed distance (in Da) between two identified peaks. This prevents the algorithm from picking multiple false peaks on the shoulders of a single broad peak.

When all the high mass deconvolution parameters have been set, the deconvolution can be triggered by activating the **Centroidation and deconvolution⇨Automatic processing ⇨High mass deconvolution** menu from the "hamburger" menu of the composite plot widget that contains the input mass spectrum. One such input mass spectrum is shown at Figure 81 – Input mass spectrum of a fluorescent protein.

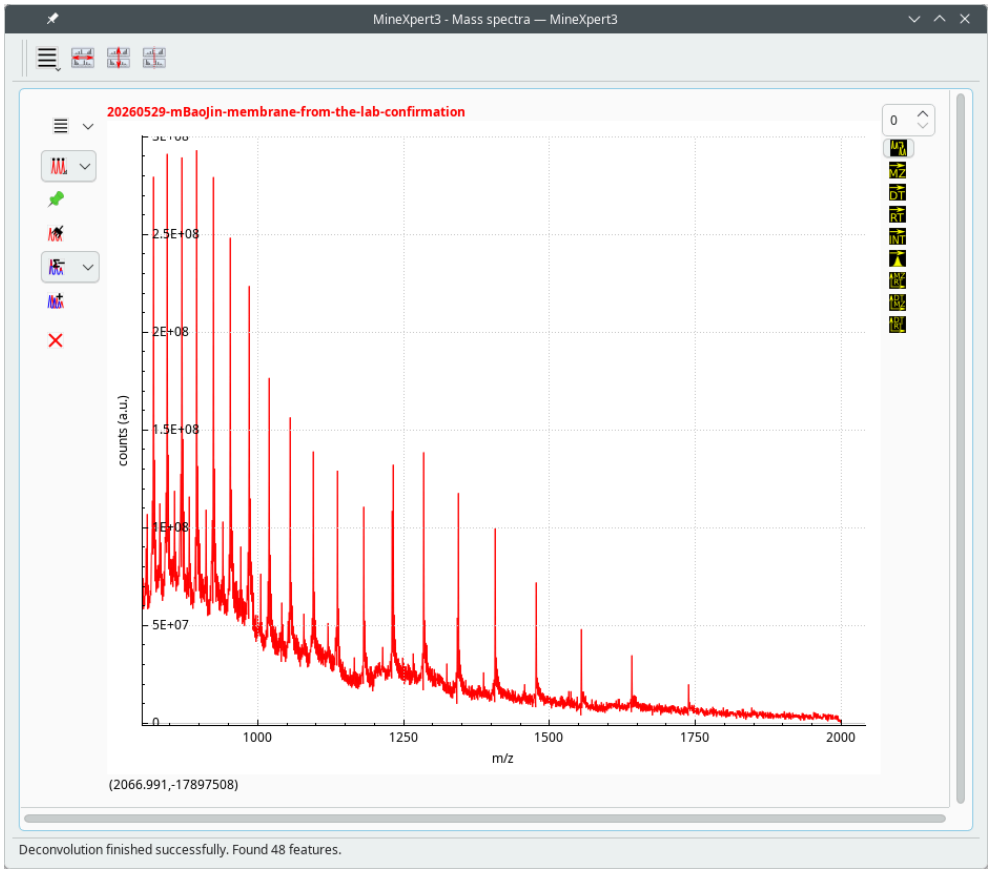


Figure 81: Input mass spectrum of a fluorescent protein

This mass spectrum was obtained on a *QExactive Plus Orbitrap instrument* at 35000 norminal resolving power. The actual resolution of the spectrum was roughly 12 k, for a protein weighing roughly 29523 Da. High mass deconvolution was performed on this mass spectrum using the parameter set shown at Figure 80 – High mass deconvolution parameter settings.

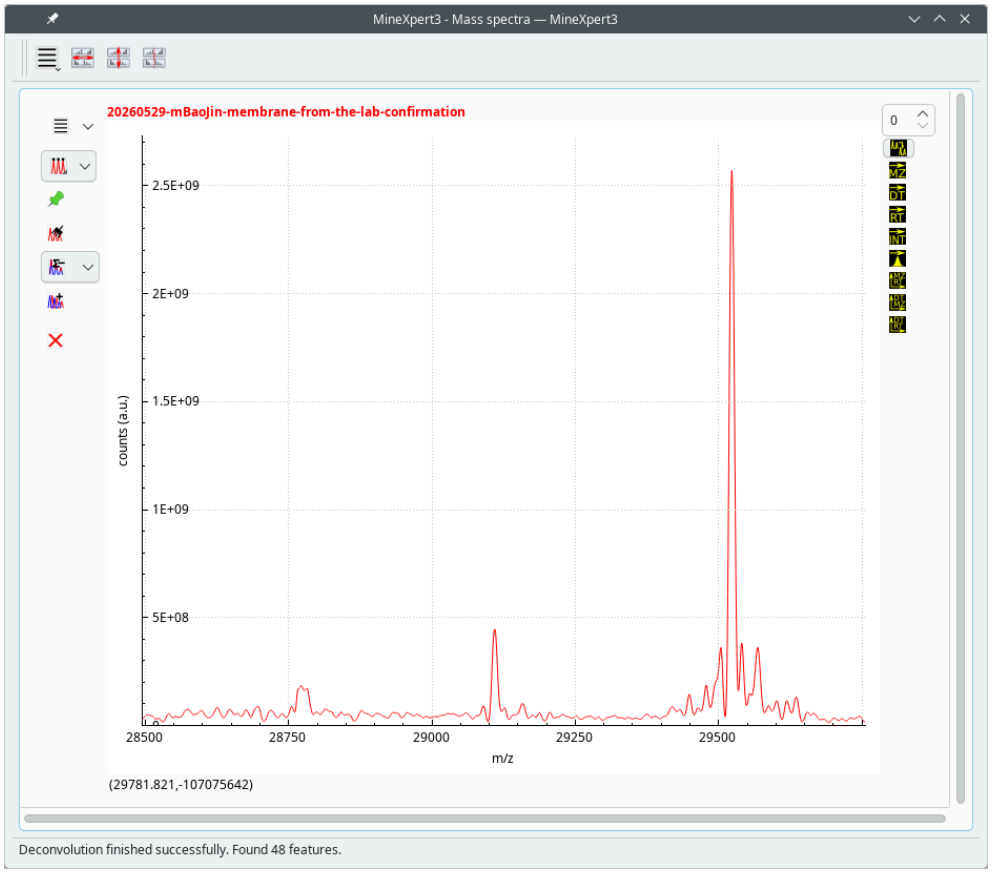


Figure 82: Deconvoluted mass spectrum of a protein

The deconvoluted mass spectrum show a number of features that may be filtered out by ajusting the high mass deconvolution pameter set.

7.5 Reading the resolving power based on mass spectral data

When ⇳-click-dragging the mouse cursor between two mass spectrum locations of interest, the program computes the apparent resolving power. This process is shown on Figure 83 – Calculation of the resolving power, where the resolving power is calculated by dragging the mouse cursor from one edge of a peak to the other at half maximum height (this is called *full width at half maximum* [FWHM] resolution).

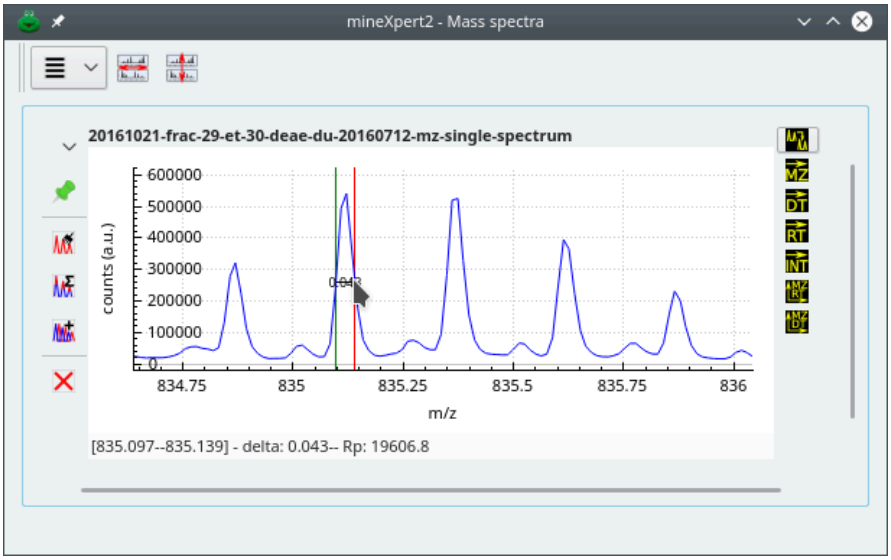


Figure 83: Calculation of the resolving power

Click-dragging the mouse cursor will trigger the calculation of the resolving power of the instrument. That value is printed in the status bar.

8 Analyte identifications

After a successful deconvolution step, nothing is more natural than to try to get help identifying the analytes under the mass peaks. Because the general process leading to analyte identification is rooted in the deconvolution of the input mass spectrum, the configuration of the analyte identification is started by checking the Check to perform analyte identification check box as shown in Figure 78 – Parameter set governing the low mass deconvolver.

Checking the box automatically displays, in the same application preferences window, the Analyte Identification page, as in Figure 84 – Configuration of the analyte identification.

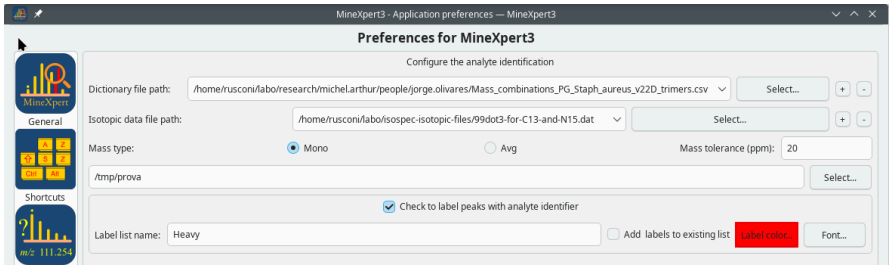


Figure 84: Configuration of the analyte identification

The analyte identification process is performed right after the deconvolution process and uses a dictionary that maps neutral chemical formulas with analyte identities.

The format of the dictionary file is CSV, with the field separator set to `Tab` (also known as the TSV format) and looks like the following. The compulsory header line must contain at least the three column headers “Id”, “Structure” and “Formula”. The “Id” column will be used to annotate the spectra.

Id	Structure	Formula
Gly	Gly	$C_2H_5N_1O_2$
Ala	Ala	$C_3H_7N_1O_2$
Glu	Glu	$C_5H_9N_1O_4$
DAP	DAP	$C_7H_{14}N_2O_4$
DAP-Ala ⟶ Glu-DAP	DAP-Ala ⟶ Glu-DAP	$C_{22}H_{38}N_6O_{11}$

Warning

The formulas listed in the dictionary file must be for neutral analytes.

> **Note**
>
> The reason why the isotopic data file path name is required is that each elemental composition of the dictionary is used to compute the mass of the analyte. This is useful, as shown in the figure, because the analytes could have been fully labelled with stable isotopes of any chemical element entering in their chemical composition.

A general overview of the centroid extraction⟶deconvolution⟶analyte identification workflow is illustrated at Figure 85 – Workflow leading from a mass spectrum to analyte identity suggestions.

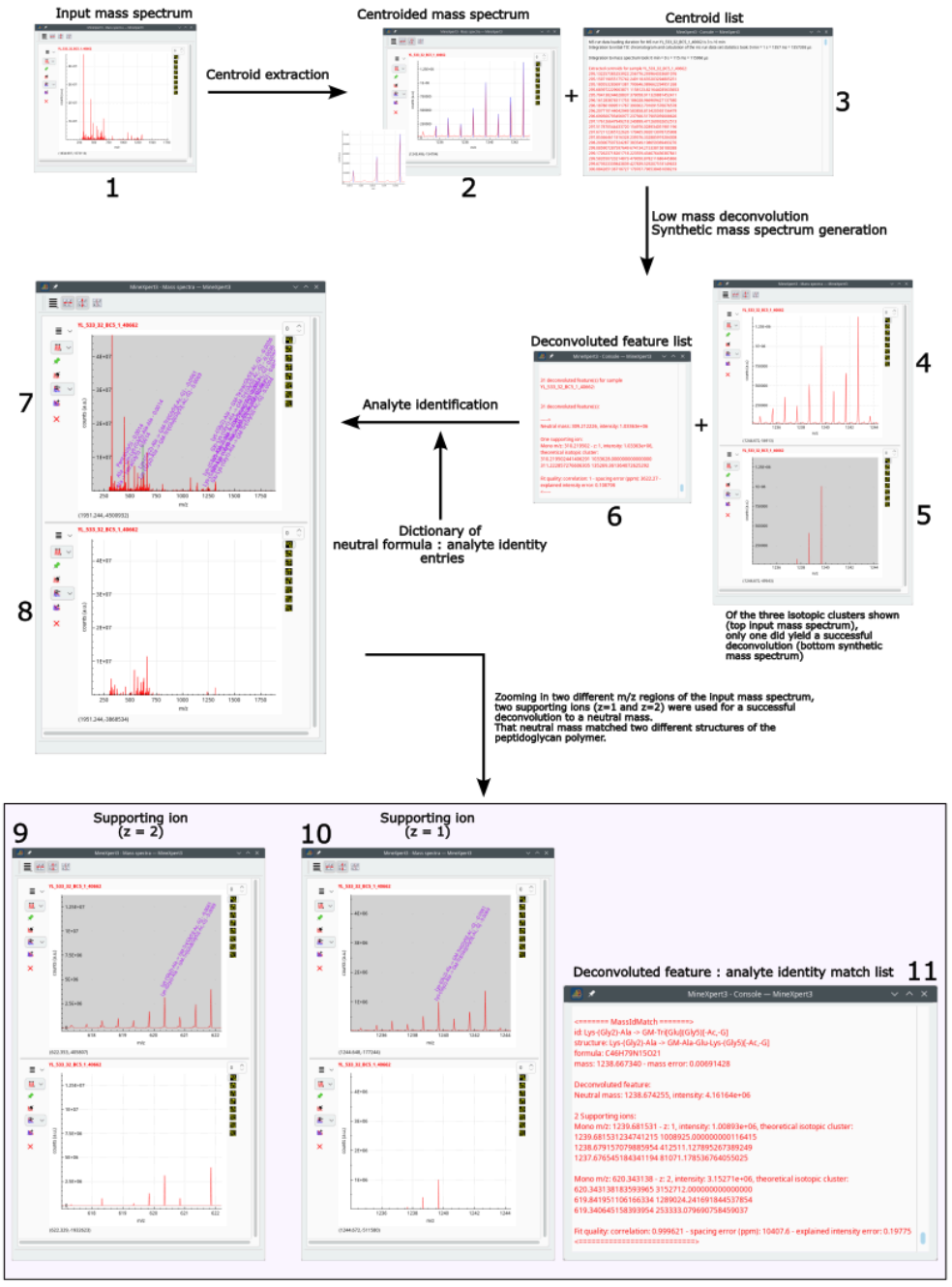


Figure 85: Workflow leading from a mass spectrum to analyte identity suggestions

Starting from a mass spectrum, it is possible to extract centroids, perform the deconvolution based on these, create a synthetic mass spectrum of the deconvoluted features and finally suggest analyte identities for these.

The figure illustrates the various steps in the workflow:

- The original mass data are integrated to a mass spectrum.
- The centroid extraction from the original input mass spectrum (1; see Section 6 – Centroid Extraction from Mass Spectra) produces a bar graph plot (blue, 2) and a list of centroids printed to the console window (3).

← *Analyte identifications*

The Processing Flow in a Data Exploration Session →

- The centroids are used to perform the deconvolution (see Section 6 – Centroid Extraction from Mass Spectra), which produces a list of deconvoluted features (6; neutral masses); each deconvoluted feature lists so-called *supporting ions* that correspond to the one or more *m/z* values that have been found to correspond to different ionization levels of that neutral feature. The signal intensity for a given deconvoluted feature corresponds to the sum of the intensities of the monoisotopic centroids in the clusters of the different supporting ions. Each supporting ion carries with itself the calculated (theoretically expected) isotopic cluster that was found to match the observed isotopic cluster for it (with calculated cluster centroid intensities normalized to the intensity of the observed monoisotopic centroid). That isotopic cluster is then leveraged in case the user requires a synthetic mass spectrum (5) that simulates a mass spectrum containing all of the deconvoluted features.

> **Note**
>
> One major interest in the crafting of the synthetic mass spectrum is to very easily check which original input centroided mass spectrum's features did actually convert into deconvoluted features. This is visible in the figure, where, in the synthetic mass spectrum (5) one isotopic cluster only was successfully deconvoluted, while the input mass spectrum (4) had three isotopic clusters.

The list of deconvoluted features is output in the console window (6).

- If the user requires help in the identification of the analytes below the deconvoluted features, they may provide a dictionary that maps analyte identifications to analyte elemental compositions (in neutral form). In that case, the software goes on and tries to map the deconvoluted features' neutral masses to the masses in the dictionary. The lower pane of the figure shows one deconvoluted feature (11) that was successfully matched to two analyte structural hypotheses.
- The left pane (9) shows the input mass spectrum (upper graph) annotated with two structural hypotheses pointing to a twice-charged isotopic cluster (z=2) that is one of the two supporting ions of the deconvoluted feature (11). The lower graph shows that in the *m/z* region, there were two deconvoluted features, but only one could be annotated. The right pane (10), shows the annotation for the supporting ion that is charged once (z=1).

Deconvolutions are performed in order to get back to the M_r mass of the analyte while reading *m/z* values. In the following sections, all the available deconvolution processes are described.

Before delving into the deconvolutions, it is necessary to present two menu options that are found in the plot widgets contained in the Mass spectra window: the menu items under the Centroidation and deconvolutions menu (Figure 68 – Mass spectrum plot widget-specific deconvolution menu).

9 The Processing Flow in a Data Exploration Session

The mass spectral data sets that are open in a given data exploration session are identifiable by the color of the traces. All the trace plots or colormap plots obtained for a given mass spectral data set are distinguished from the others by the color used for the plotting (trace or bars plots) or for the axes and labels (colormap plots). The color is thus a means to maintain a visual "filiation" between plots. But for the various integration steps of a mass spectral data exploration session involving the same mass spectral data set, there needs to be a way to record them in a detailed manner. The means by which this is achieved is the subject of this chapter and is based on the recording of all the integration steps (*ProcessingStep* records) into a general record that is called *ProcessingFlow*.

9.1 Recording Processing Steps Linked to the Initial Mass Spectrometry Dataset

One introduction about the *ProcessingFlow* concept is at Section 5.1.1 – Processing Flow Entities Document all Integrations.

The general notion of *ProcessingFlow* is that from the opening of a mass spectrometric data file to the end of the data exploration session, all the integration steps elicited by the user of *MineXpert* are recorded in such a way that the program will be able to run them all in memory when new integrations are performed.

Let's take an example: the user works with ion mobility mass spectrometry data and is exploring the efficiency of ion mobility separation of ions of the same molecule (say, ubiquitin, for example) as a function of a number of instrument setting parameters that are modified during the acquisition.



Note

The dataset used for Section 5.6 – Chained Integrations and this section was kind courtesy of Dr Valérie Gabelica and correspond to a work entitled *Optimizing Native Ion Mobility Q-TOF in Helium and Nitrogen for Very Fragile Noncovalent Structures* published in *JASMS* with DOI: 10.1007/s13361-018-2029-4. Many thanks to Dr Valérie Gabelica (at the time at IECB, Bordeaux, France) for permission to use her data set.

For the first two minutes of the acquisition (min [1-2]), the instrument parameters tend to let the ubiquitin adopt a more regrouped, or condensed, tridimensional shape. The ions will thus have a drift time corresponding to that shape (see below, where this is demonstrated). During the next two minutes of the acquisition duration (min [3-4]), the instrument parameters are modified to induce a gradual defolding of the oligopeptide, which leads to an increased drift time compared to that observed in the first two minutes of the data acquisition. Three more such steps are carried out (see below).

Now, the user wants to precisely quantitate the ions that arrive to the detector during these five acquisition time ranges so as to quantitatively compare the ion mobility-based separation of the ubiquitin ions as a function of the different instrument settings.

There is one more consideration to this data exploration session: the ubiquitin molecule is ionized in the [2+–8+] range and the user wants to only compare ion counts for the 6+ charge level.

What would be the sequence of ProcessingStep events in the exploration to achieve what the user wants? This is what is described in the text below.

- Obtain the whole TIC chromatogram from the loaded mass spectrometry data file.

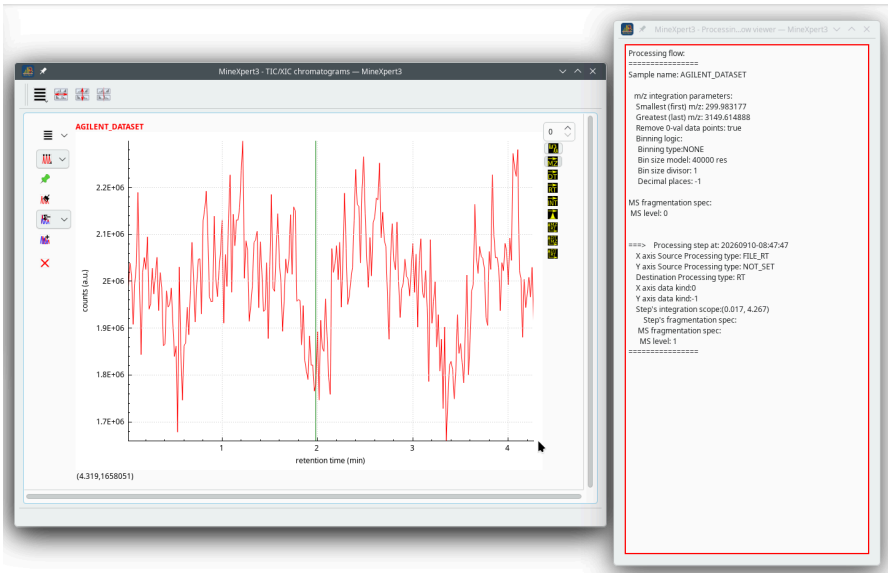


Figure 86: TIC chromatogram

The first step is about reading a file and processing the TIC chromatogram (source processing type is FILE_RT, with "RT" standing for "retention time") to actually plot it to a TIC chromatogram (destination processing type is RT). Note how specific integration ranges are recorded also, which is of utmost importance if these ProcessingStep records are then used to "replay" the integrations later.

- Integrate the whole TIC chromatogram to a mass spectrum.

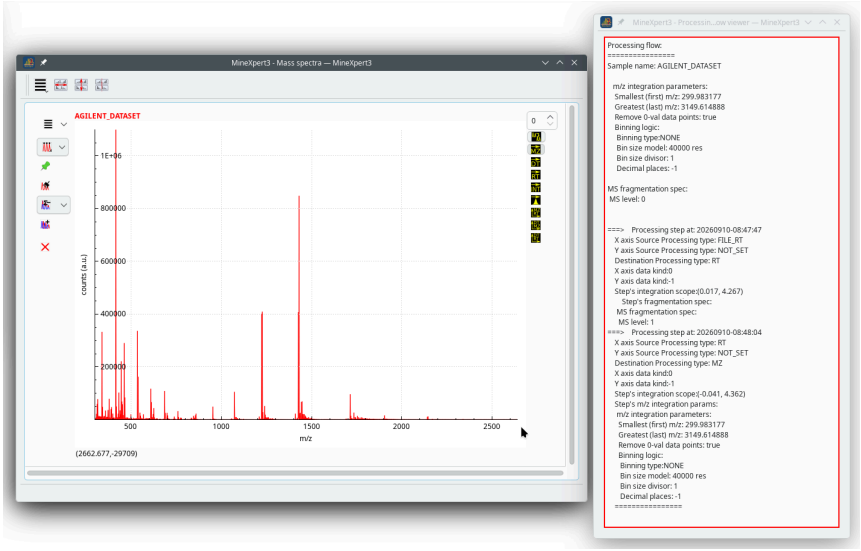


Figure 87: Mass spectrum

The second step is about integrating the whole TIC chromatogram (source processing type is RT) to a mass spectrum (destination processing type is MZ) to plot it in the mass spectrum plot window. Again, note how all the relevant metadata are stored. In particular, the binning mode is NONE because the data from the Agilent instrument need no binning, and we'll see how beautiful these data are.

- Inspect the mass peaks and find the isotopic cluster that corresponds to charge level 6+.

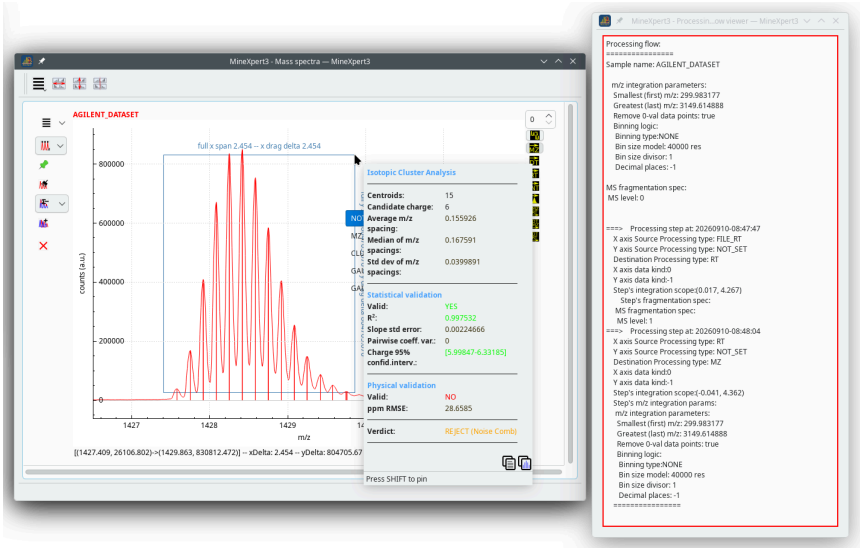


Figure 88: Isotopic cluster of the 6+ ion

Now, the user needs to explore the mass spectrum's data and find the proper 6+ ion. Use the charge determination feature (Section 7.2.3 – Multi-point statistical analysis-based charge determination and mass estimations) to unambiguously assess the charge of the ion beneath the isotopic cluster at *m/z* 1428.

- Integrate that individual isotopic cluster to an ion mobility (drift) spectrum.

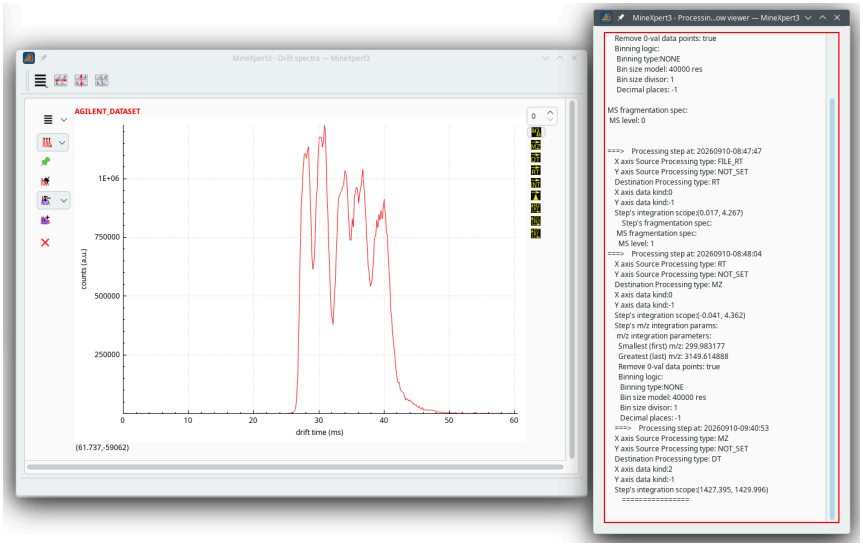


Figure 89: Ion mobility (drift) spectrum

The third step is the integration of the whole isotopic cluster singled out as for a 6+ charge level ion to an ion mobility (drift) spectrum. As visible, there are five ion mobility peaks delineating five ion mobilities for the same molecular species with the exact 6+ ionization state. These ion mobilities are due to the instrument settings that were tuned by the authors of the research paper.

- Integrate to intensity the five observed ion mobility peaks.

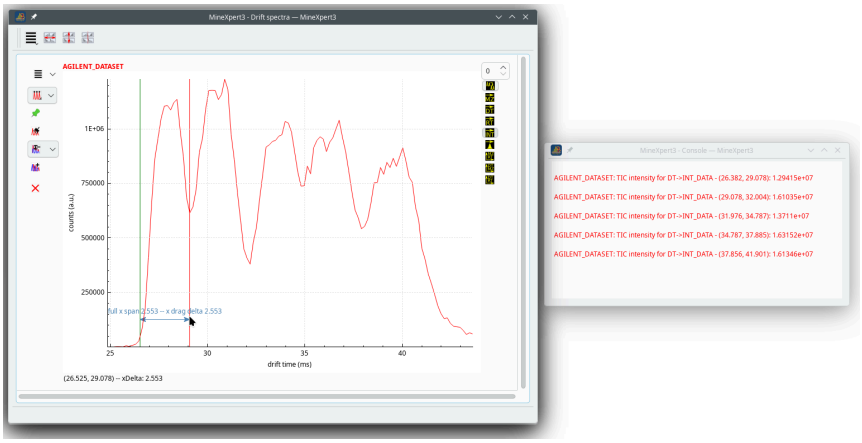


Figure 90: Intensities are determined for the various ion mobility peaks

At this point, each ion mobility peak "contains" the ions that match what we are searching for: ions from the whole mass spectral acquisition dataset that correspond to ion charge level 6+ of ubiquitin (and not any contaminant that might have shown in the full mass spectrum) and that show varying ion mobilities because of the settings used by the authors to modify the mass spectrometer behaviour. Indeed, the integration from mass spectrum to ion mobility spectrum was performed *only* for the isotopic cluster of [ubiquitin + 6H]$^{6+}$.

This is the reason why integrating each one of the ion mobility spectrum's peaks to an intensity (by selecting the INT button on the right hand side of the plot widget) gives the *precise* ion count of the selected ions. The results of the five integrations are provided in the Console window shown right of the plot window. The first integration is for ion mobility peak at drift time range [26–29] ms.

One now needs to confirm that each ion mobility mass spectrum's peak actually corresponds to specific mass spectrometer settings as described in the introduction of this section. Achieving that is very simple: select the RT button on the right hand side of the plot window and integrate to a TIC chromatogram each individual peak (starting from left peak at ion mobility drift times in the [26–29] ms range) up to the last one on the right. The result (Figure 91 – Retro-integrations from ion mobility spectrum peaks to TIC chromatograms) shows that indeed the different ion mobility peaks correspond to acquisition time ranges that are neatly discriminated by the integrations.



Figure 91: Retro-integrations from ion mobility spectrum peaks to TIC chromatograms

The first chromatogram is the full acquisition's mass data file's TIC chromatogram and all the chromatograms below are the results of the retro-integration from each ion mobility peak to the corresponding TIC chromatogram (which, in the end, could be called XIC chromatogram, for *extracted ion current* chromatogram).

As shown in the sequence of images above, each time a new integration is performed, a corresponding ProcessingStep record is added to the general ProcessingFlow object. This is visible each time in the text that is displayed in the small window right of the plot window.

So, what is all this demonstration about? Well, the question is, how were the intensities calculated, as shown in Figure 90 – Intensities are determined for the various ion mobility peaks? To answer this question we need to recall what happened at each new step of the

ion mobility mass spectrometric data exploration: a new ProcessingStep record detailing all the bits of the integration was added to the ProcessingFlow object that is systematically attached to any plot in the program. Each plot that is plotted as a result of an integration has thus a corresponding ProcessingFlow object that it inherited from the plot at which the integration took place.

This is visible by reading the ProcessingFlow text shown in the figures above and by ascertaining that new ProcessingStep records are added to existing ones. This feature of adding to the ProcessingFlow a new ProcessingStep record for each new integration fully documents the history path that leads to any plot in the program, starting right from the very first step, that is, the reading of the data from the file on disk.

When the user integrates one ion mobility spectrum's peak to INT, the program gets the ProcessingFlow for the plot and plays—in memory— all of its ProcessingStep records right from the full data set that was initially read from disk. Playing all the steps one after the other effectively restricts the scope of the initial mass dataset to a limited subset. The set of ProcessingStep records contained in the ProcessingFlow thus acts as a multi-component data filter.

In the end, all the mass data that could "pass" all the ProcessingStep records have their ion count summed up to the intensity value that is displayed as a result of the integration.

9.2 Recording Processing Steps Detached from the Initial Mass Spectrometry Dataset

In the previous section, we saw how ProcessingStep records are added to the ProcessingFlow object that documents the history of any given plot in *MineXpert*. In the examples above, all the integrations did involve the initial dataset and subsets of that dataset as documented by the ProcessingStep records, such that it is effectively possible to replay all the ProcessingStep records from the first one to the last one: *the logical link between the various plots and the initial dataset as read from disk has never been disrupted*.

There are situations where this stops from being true. And this is what this section is all about.

Let's take an example based on the same dataset as that used in the previous section. At one moment (Figure 88 – Isotopic cluster of the 6+ ion) the user wanted to ascertain the charge of the ions beneath a given isotopic cluster. That process involves extracting and plotting the centroid of each isotopic clusters' peak that is selected by the mouse dragging movement.

The centroids bars plot that is plotted into the same plot widget that the user is selecting in Figure 92 – A centroids bars plot is no more logically linked to the initial mass dataset, is no more linked to the initial mass spectrometric data set.

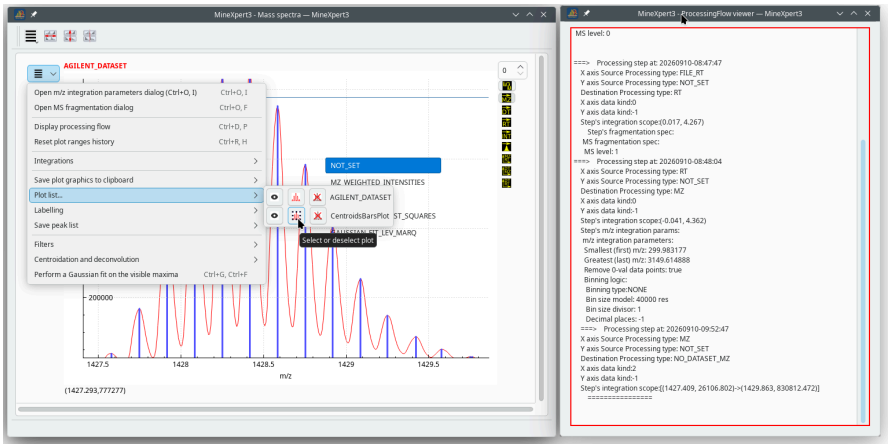


Figure 92: A centroids bars plot is no more logically linked to the initial mass dataset

Because the centroids bars plot is a pure creation of data and not the result of a conventional initial dataset-based integration, its data are no more logically linked to the initial mass spectrometry dataset (they are detached). The ProcessingFlow's last ProcessingStep record shows this as an integration from source type MZ to destination type NO_DATASET_MZ.

What does the integration destination type NO_DATASET_MZ do in the context of the use of the centroids bars plot later? The presence of NO_DATASET_MZ as either source or destination integration type stops *MineXpert* from trying to use the initial dataset when actions are requested on that centroids bars plot. Let's say that the user wanted to integrate the centroids bars plot to an intensity (INT) or a mass spectrum (MZ) for later use (whatever that later use may be). The user would select the centroids bars plot and then check the MZ button on the right hand side of the plot widget and perform a $^{0}_{2}$-click-drag operation over the centroids bars plot. *MineXpert* would interpret that as a normal integration, look for the *m/z* range covered by the mouse drag movement and try to replay all the ProcessingStep records in the ProcessingFlow. If there was no ProcessingStep record documenting that the centroids bars plot is no more logically linked to the initial mass dataset, the result of that last integration would be obtained for the isotopic cluster graph plot displayed in the figure, not for the centroids! Which would of course mean nothing.

The ProcessingStep record listing NO_DATASET_MZ as an integration source and/or destination type allows for interesting computing possibilities. For example, when an integration such as that mentioned above is tried, the program checks the compatibility between the source plot data which is, in this example, the centroids bars plot (source type NO_DATASET_MZ) on the one hand and the integration destination type, which is either a mass spectrum (MZ) or an intensity (INT). The result of check is positive because the centroids bars plot is indeed *m/z* data (source type NO_DATASET_MZ) and also an intensity

can be readily computed from the plot itself. In this case, the integration to a mass spectrum would work: the new plot would be a centroids bars plot exactly as the parent plot was a centroids bars plot. In this case, the data of the plot itself is used to create the requested artifact: a new MZ or a INT intensity value to be printed in the Console window.

Let's imagine that the user did not want to integrate the centroids bars plot to a mass spectrum but to a drift spectrum. The program cannot comply: how can a source plot of type MZ be used to plot a destination plot of type drift spectrum (DT) without needing the initial mass spectral dataset, exactly as performed in Figure 89 – Ion mobility (drift) spectrum?



Tip

When performing integrations starting with plot data that have no more a logical link to the initial mass spectral dataset, the program only allows performing the requested integration either:

- If the integration destination type is the same as the source integration type (NO_DATASET_DT to DT, NO_DATASET_MZ to MZ, NO_DATASET_RT to RT), in which case the source plot data are used for the integration;
- If the integration destination type is INT intensity value. In this case also, the source plot data are used for the computation.

10 Recording data exploration discoveries

When doing mass analysis work it is often desirable to store the painstakingly manually picked *m/z* or M_r values for later use. *MineXpert* provides a flexible process allowing one to record the data discoveries during the exploration of a mass data set to any combination of the following destinations: file, clipboard, console.

In order to record the innumerous analysis steps that result from a data exploration session, the configuration of the discoveries storage must have been done by accessing **Preferences⇨Edit preferences...** and then selecting the Discoveries section, as described in Figure 5 – Application preferences - Discoveries page.

In that window, the user can select the destination of the data analysis recording system: console, clipboard, file or any combination of the three. When electing to record the discoveries to file, the user might specify if the recording would overwrite any preexisting file or, instead, append to that file. Depending on the kind of plot where data exploration occurs, the format of the data to be recorded needs to change. Indeed, it would make no sense to record the charge *z* when exploring data in the Drift spectra window. This is why the text format of the data export needs to be defined for each one of the three kinds of plots: TIC/XIC chromatogram, mass spectrum or drift spectrum (Specialized formats group box).

Before delving into the configuration intricacies, let us tell immediately how to trigger the recording of the exploration discoveries: using the `Space` bar in the composite plot widget containing the graph being analyzed.

The text format specification can contain particular tokens as described below.

> **Note**
>
> Almost all the format string elements below are numeric (representing either integer or float values). For float values, it is possible to define the number of decimals to be used when printing the value. The decimals specifier is of the form ".", like ".3".
>
> When used with a format string, that gives "%X.3", which specifies that when printing out the x-axis value, 3 decimals should be added after the decimal point. To prevent decimal at all, do not use any decimal specifier at all ("%X").

- %f: mass spectrometry data file name.
- %s: sample name.
- %X: value on the x-axis of the graph (no unit). For a drift spectrum, that would be ion mobility times, for a mass spectrum, that would be *m/z* values, for a TIC/XIC chromatogram, that would be retention times.
- %Y: value on the y-axis of the graph (no unit). This value corresponds to the value at the cursor point (not at a point on the curve). In all the graph plots, that would be intensities in any unit provided by the mass spectrometer (typically, counts).
- %C: value on the y-axis of the graph (no unit). This value corresponds to the value at the curve of the selected trace plot (not at the cursor point). In all the graph plots, that would be intensities in any unit provided by the mass spectrometer (typically, counts).
- %x: delta value on the x-axis (when appropriate, no unit).
- %y: delta value on the y-axis (when the mouse has been dragged enough to draw a rectangle). The delta value is computed as the difference between the y-axis value at the mouse drag start point and the y-axis value at the mouse drag stop point.
- %c: delta value on the y-axis (when the mouse has been dragged enough to draw a rectangle). The delta value is computed as the difference between the y-axis value *on the curve* at the mouse drag start point and the y-axis value *on the curve* at the mouse drag stop point.
- %I: TIC intensity after TIC integration over a graph range.
- %b: x-axis range *begin* point for the computation (where applicable, for example for the TIC integration to a single value).
- %e: x-axis range *end* point for the computation (where applicable).
- *For mass spec plot widgets:*:
  - %z: charge.
  - %M: M_r as computed during deconvolution (see Section 7 – Mass Spectral Deconvolutions).

> **Note**
>
> It is important to keep in mind that the %z and %M format strings can only work if the user is actually analyzing a mass spectrum and if the user has *effectively* performed a deconvolution operation that has allowed computing these two values. If the values are not available, the program shows the format string untouched ("%M", for example) in the textual output upon hitting the Space bar (see below). Likewise, the %I format string will only hold meaningful

data if an integration to a single TIC intensity value has been determined (see Figure 49 – Integrating mass data from a mass spectrum to a single TIC intensity value using the processing flow) for that specific data analysis record.

The recording of data analysis steps works in any trace plot widget (not yet in the colormap plot widgets) even if there are more than one trace in a given plot widget. In the case the plot widget has more than one trace and none or more than one is selected, the program will ask that a single trace be selected. If data are to be scrutinized for more traces, simply select each trace in turn and trigger the analysis stanza to be output using the Space bar each time.

Once configured, the format strings might be stored in a drop down box for later use. To that end, click onto the Add to history button while having the format text displayed in the text editor and it will be appended to the drop-down list. The list gets stored when the dialog window is closed and will be filled-up again when the program is restarted.

As an example, if the user defined the following format string for a mass spectrum:

Formatting specification for the data discovery stanza

```
Mass spec. :
mz = (%X, %Y) z = %z
filename = %f
date = 20161021
session = 20161021
mslevel = 1 msion = esi msanal = tof
chrom = DEAE fraction = 25
seq =  pos =     oxlevel = 0 pos =
intensity =
comment =
```

Then, a resulting data exploration stanza that would be recorded will look like this:

Output data discovery stanza

```
Mass Spec. :
  mz = (1051.8, 50863) z = 1
  filename = 20161017-rusconi-frac-25-deae-20160712.mzml
  date = 20161021
  session = 20161021
  mslevel = 1 msion = esi msanal = tof
  chrom = DEAE fraction = 25
  seq =  pos =     oxlevel = 0 pos =
  intensity =
  comment =
```

Interestingly, the user can define any kind of format, leaving fields available for later filling-in. This feature is of immense value when the analysis file is used later to fill-in a database for easy storage and interrogation of the exploration discoveries. In this case, it would be useful to have the file opened in an editor and at each new stanza, edit the comment field if something needs to be commented, like the shape/intensity of a mass peak, for example.

Note that the program closes the file each time a new stanza has been written. This makes it possible to edit that file safely in between each stanza record. Remember to force the editor to reload the file from disk after each exploration discovery recording.

When the recording involves sending the analysis data to the console, the data are sent to it as text that is colored the same as the spectrum that was under scrutiny.

When the mouse cursor has been placed at the proper location on the graph (with or without Φ-click-dragging, depending on the situation), the user hits the Space bar and the data analysis stanza is recorded to the selected destination(s): console, clipboard, file.

Appendix

GNU General Public License version 3

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. https://fsf.org/

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The *GNU* General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the *GNU* General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the *GNU* General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the *GNU GPL* protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the *GPL* clearly explains that there is no warranty for this free software. For both users' and authors' sake, the *GPL* requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the *GPL* to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the *GPL*, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the *GPL* assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the *GNU* General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified
version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes

copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal
Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting User's Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

1. The work must carry prominent notices stating that you modified it, and giving a relevant date.
2. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
3. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
4. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

1. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
2. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
3. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
4. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
5. Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a

product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in *ROM*).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part

of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

1. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
2. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
3. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
4. Limiting the use for publicity purposes of names of licensors or authors of the material; or
5. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
6. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2)

arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Other's Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the #acronym(GNU) Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the *GNU* Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the *GNU* Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the *GNU* General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the *GNU* General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the *GNU* General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the *GNU* General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFEC-

TIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

> *one line to give the program's name and a brief idea of what it does*
>
> Copyright (C) *year — name of author*

> This program is free software: you can redistribute it and/or modify it under the terms of the *GNU* General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.
>
> This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the *GNU* General Public License for more details.
>
> You should have received a copy of the *GNU* General Public License along with this program. If not, see https://www.gnu.org/licenses/.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

> *program* Copyright (C) *year — name of author*
>
> This program comes with ABSOLUTELY NO WARRANTY; for details type "`show w`". This is free software, and you are welcome to redistribute it under certain conditions; type "`show c`" for details.

The hypothetical commands "`show w`" and "`show c`" should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the *GNU GPL*, see https://www.gnu.org/licenses/.

The *GNU* General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the *GNU* Lesser General Public License instead of this License. But first, please read https://www.gnu.org/licenses/why-not-lgpl.html.

Colophon

About the authors

Filippo Rusconi is a senior research scientist at the French national research council (*Centre national de la Recherche scientifique*, *CNRS*). Filippo has a background in biochemistry and organic chemistry and was trained during his Ph.D. as a bioanalytical chemist. He has extensive knowledge of analytical techniques involved in the study of biopolymers.

Filippo Rusconi is the author of a handbook about mass spectrometry for biochemists (French). The book was published by the French sci/tech publisher Lavoisier Tech&Doc.

Book preparation

The look of this book (*PDF* file) is the result of me having read many books from the O'Reilly publisher.

The typesetting of the book has been done on a Debian GNU/Linux computer using only Free Software. The *Typst* application was used for rendering the document code.

The main font used is EBGaramond and the symbol/mathematical font was from the STIX project (font: STIX2Math).

The screen shots were taken with Spectacle, the screen capture program shipped along with my KDE desktop environment and resampled either with *Gimp* (the GNU image manipulation program) or *Krita*. Illustrations were done in Inkscape, a vectorial drawing software.

Index

A

Acquisition
Accumulation mode 26
Profile mode 27
Analyte
Identification 142
Centroid extraction 145
Deconvoluted feature 145
Dictionary 142
Process overview 144
Application
Preferences 36
General options
Decimals 37
Thread configuration 36
Shortcuts 38
Axis
Locking views 74

B

Binning
Bad bin size setting 83
Computation time 89
Finding the right bin size 84
Mass spectrum resolution 88
Setting the m / z integration parameters 83

C

Centroid
Extraction 68, 110
Analyte identification 145
Configuration 110
Signal-to-noise ratio 110
Plotting of extracted centroids 112
Citation 26
Combination
Spectrum 27
Composite plot widgets
Alignment 75

Colormap plots 64
Integrations 96
Menu 69
Transpose axes 70
Z axis transformations 70
Graph and bars plots 64
Integrations 91
Menu 64
Locking views on axes 74
MS^n Selector 81
Vertical rule marker 75

D

Data
Diversity from vendors 94
Exploration
Mouse driven methods 72
Statistical analysis 95
Decimals 37
Deconvolution 68, 114
Automated 145
Charge state envelope
Charge and Mr determination 115
Feature 145
High mass 135
Isotopic cluster
Multi-point
Charge determination 120
Charge determination with mass estimation 127
Two-point
Charge and Mr determination 118
Low mass automated on isotopic clusters 130
Configuration 131
Discovery
Recording 38, 156
Configuration 156
Format string 157

F

Feature
Alignment vertical rule marker 75
Filters
Colormap plot high-pass filter Z axis 72, 99
Colormap plot low-pass filter Z axis 71, 99
Colormap plot Z-axis log10 filter 71
Savitzky-Golay 68, 89

Configuration 90
Fit
Gaussian 68, 127
Levenberg-Marquardt 127
Fragmentation
Level 79
MS^n 79
Precursor index 80
Precursor m / z 80

G

Gaussian 68, 127
Git repository 34
Graphical interface
Composite plot widgets 64
Main data plotting windows 48

I

Integrations 77
Chaining integration steps 105
Destination type selection 62
From colormap plots 96
From graph and bars plots 91
Integration data reception 60
Involving multiple widgets at once 107
ProcessingFlow 77
Setting the m / z parameters 81
TIC->DT 30
TIC->MZ 29
To a drift spectrum 92
To a mass spectrum 91
To a TIC chromatogram 92
To a TIC intensity
Using the dataset 92
Using the plot data points 93
With summing intensities 108
Ion mobility
Flattening IM scans 41
Keeping IM scans 41
Isotopic data 133

L

Labels
List 67

Levenberg-Marquardt Gaussian fit 127
License 160
Locking axes between plots 74

M

M / z
  Binning 82
Mass spectrometry
  Data
    Loading from file 40
    Open datasets 45
    Table view 45
MsXpertSuite
  History 22
  MassXpert 20
  MineXpert 20

P

Panning plots 74
Plot
  List all 66
  Save 66
  Save peak list 67
Preferences 36
ProcessingFlow 65, 147
  Detached from initial MS dataset 153
  ProcessingStep 77, 78, 147
  Replay steps 78

R

Resolving power
  Read value from spectrum 141

S

Savitzky-Golay 68, 89
  Configuration 90
Shortcuts
  Configuration 38
Spectrum
  Combination 27

T

Thread configuration 31, 36
TIC 28
Total ion current chromatogram 28

Z

Zooming in/out plots 73
  Recording zoomed view history 74
  Zooming-in 73
  Zooming-out 74