



## MZmine 3

---

**Documentation**

*The MZmine Community*

*None*

## Table of contents

---

<b>1. Welcome to the MZmine 3 wiki!</b>	4
<b>1.1 What's new compared to MZmine 2?</b>	4
<b>1.2 About this documentation</b>	4
<b>1.3 How to contribute</b>	4
<b>2. Getting Started</b>	5
<b>2.1 Download</b>	5
<b>2.2 Installation</b>	5
<b>2.3 Set User Preferences</b>	8
<b>3. Main window overview</b>	9
<b>3.1 MS data files and feature lists tab</b>	9
<b>3.2 Main content pane</b>	9
<b>3.3 Main menu</b>	9
<b>3.4 Task overview</b>	9
<b>3.5 Page Contributors</b>	10
<b>4. Visualization modules</b>	11
<b>4.1 Visualization modules</b>	11
<b>4.2 MS data visualisation</b>	12
<b>4.3 Ion mobility raw data overview (LC-IMS-MS)</b>	16
<b>4.4 Image viewer</b>	18
<b>4.5 Processed data visualition</b>	19
<b>5. Processing modules</b>	27
<b>5.1 Data import</b>	27
<b>5.2 Mass detection</b>	29
<b>5.3 Mobility scan merging</b>	46
<b>5.4 ADAP chromatogram builder</b>	47
<b>5.5 IMS Expander</b>	49
<b>5.6 Ion mobility trace builder</b>	50
<b>5.7 Smoothing</b>	52
<b>5.8 Local Minimum Resolver</b>	54
<b>5.9 CCS Calibration and calculation</b>	58
<b>5.10 Gap-filling</b>	61
<b>5.11 MS2 Scan Pairing</b>	62
<b>5.12 <math>^{13}\text{C}</math> isotope filter</b>	64
<b>5.13 Isotope pattern finder</b>	66
<b>5.14 Spectral library creation</b>	68

5.15 Spectral library search	69
5.16 GNPS-FBMN/IIMN export	72
5.17 Join aligner	74
5.18 Other parameters	77
6. Workflows	80
6.1 LC-MS Workflow	80
6.2 LC-IMS-MS Workflow Overview	82
<b>6.3 Batch processing</b>	85
6.4 Processing wizard	86
7. Additional resources	88
7.1 General terminology	88
7.2 MZmine-specific terminology	90
7.3 Ion mobility spectrometry terminology	92
7.4 Graphical comparison of LC-MS and LC-IMS-MS data	94
8. Performance options	95
8.1 Preferences	95
8.2 Logs	96
9. Command-line arguments	98
10. Contribute	99
10.1 How to contribute	99
10.2 Development in IntelliJ	103
11. Acknowledgements	116
11.1 Related projects	116
11.2 Libraries we use in MZmine	116

# 1. Welcome to the MZmine 3 wiki!

---

MZmine 3 is an open-source and platform-independent software for mass spectrometry (MS) data processing and visualization. It enables large-scale metabolomics and lipidomics research by spectral preprocessing, feature detection, and various options for compound identification, including spectral library querying and creation.

Since the introduction of MZmine 2 in 2010, the project has matured into a community-driven, highly collaborative platform and its functions continue to expand based on the users' needs and feedbacks. This has also enabled the tight integration of the MZmine ecosystem with popular third-party software for MS data analysis, such as the [SIRIUS](#) suite for *in silico* metabolite annotation, the [GNPS](#) platform with Ion Identity Molecular Networking, the [MetaboAnalyst](#) web app for univariate and multivariate statistical analysis, *etc.*

Such a great progress was made possible by the invaluable contribution of many [developers](#) from research labs distributed all over the world!

Want to get started with MZmine 3? Check out our [getting started](#) page!

---

## 1.1 What's new compared to MZmine 2?

---

MZmine 3 comes with a redesigned and fully customizable [GUI](#) based on the JavaFX technology that allow an interactive visualization and validation of results from every processing step.

A completely new data structure provides the flexibility to process any type of mass spectrometry, including LC-MS, GC-MS and MS-imaging. Moreover, MZmine 3 now supports ion mobility, with a dedicated [LC-IM-MS data visualization](#) module and [feature detection](#) algorithms.

Finally, significant effort was devoted to trace memory issues and bottlenecks, resulting in an unprecedent processing performance and scalability.

**COMING SOON!** We are implementing the [Mass Spec Query Language](#) (MassQL) to explore your MS data with human-readable, succinct queries! The project is supported by the [Google Summer of Code](#) program.

---

## 1.2 About this documentation

---

Here you can find documentation for both processing and visualization modules in MZmine 3. Moreover, data processing pipelines for untargeted [LC-MS](#) and [LC-IMS-MS](#) feature detection are described and general recommendations are given.

**COMING SOON!** We are currently working on a series of short videotutorials to help get you started with the main features of MZmine 3!

---

## 1.3 How to contribute

---

The MZmine community is always welcoming new developers and contributions! You can contribute by improving existing modules or even adding new features in MZmine 3! Please, check out our brief [tutorial](#).

You can also contribute to this wiki and help new users to get started with MZmine 3! See [here](#) how to contribute to the documentation.

---

## 2. Getting Started

---

### 2.1 Download

Download MZmine 3 portable versions or installers from GitHub:

<https://github.com/mzmine/mzmine3/releases/latest>

### 2.2 Installation

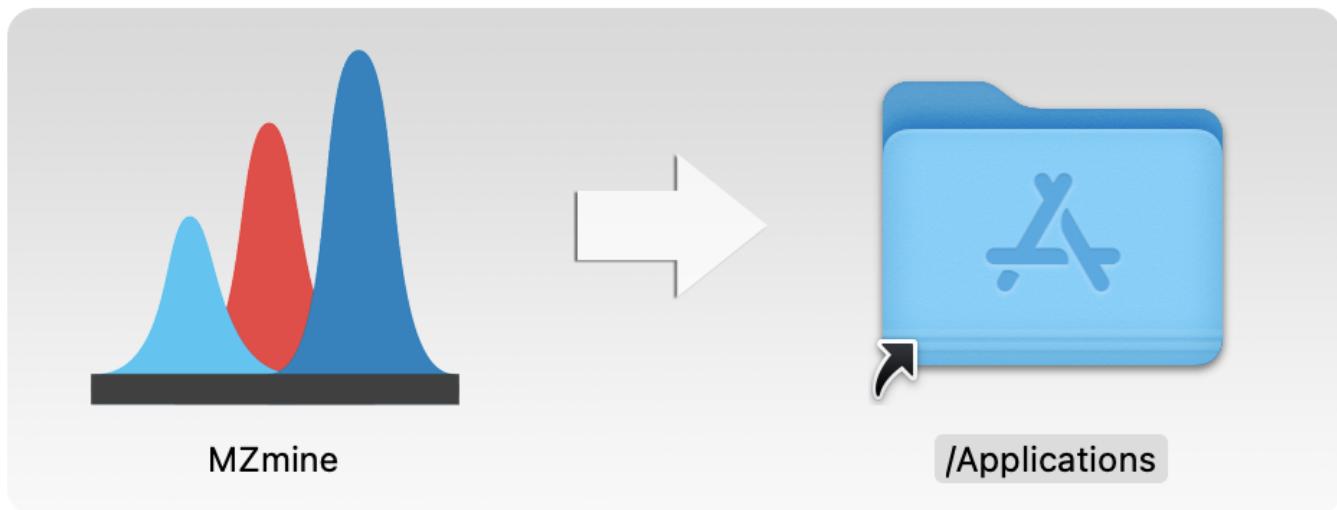
On Windows and Linux the installers and portable versions should function directly. Windows users might be warned that MZmine is not signed or from a trusted source and have to click run anyways. Before creating your first project, we recommend to [set the preferences](#).

#### 2.2.1 On macOS

Currently, MZmine 3 lacks a signature for macOS. While we are working on this, user can allow MZmine in the macOS Gatekeeper protection by running the following command in the terminal from the Applications folder.

- Download MZmine and click the MZmine.dmg installer - Drag and drop MZmine into the Applications folder
- Open the Applications folder, right click (CTRL click) anywhere, e.g., on the MZmine icon, and choose "New Terminal at folder" from the context menu
- Run the provided command to tell macOS to trust the installed version of MZmine. The terminal directory has to be the Applications folder. (Depending on the actual folder use or omit the `..` to jump to the parent directory).
- Approve command with user password
- Start MZmine

```
sudo xattr -cr ..../MZmine.app
# if this fails try
sudo xattr -cr MZmine.app
```



Open

Show Package Contents

---

Move to Trash

---

Get Info

Rename

Compress "MZmine"

Duplicate

Make Alias

Quick Look

---

Copy

Share >

---



Tags...

---

Quick Actions >

---

New Terminal at Folder

New Terminal Tab at Folder

The Terminal does not output any log or message.

```
(base) mauriciocaraballo@Mauricios-MacBook-Pro MZmine.app % cd ../
(base) mauriciocaraballo@Mauricios-MacBook-Pro /Applications % sudo xattr -cr MZmine.app
(base) mauriciocaraballo@Mauricios-MacBook-Pro /Applications %
```

Before creating your first project, we recommend to set the preferences.

## 2.3 Set User Preferences

---

Before creating your first project, we recommend setting up some things.

1. Set a temporary file directory. Go to *Project* → *Set preferences* → *Temporary file directory*. This requires a restart to take effect.
  - a. We recommend setting the directory to an SSD with enough space for fast processing and visualizations.
  - b. On Windows, old temporary files are deleted when a new session is started.
2. MZmine 2 projects cannot be imported due to changes in the data structure.
3. MZmine 2 batch files cannot be imported due to parameter optimizations.

You can get familiar with the new GUI here: [Main window overview](#)

A quick insight to data processing workflows can be found here: [LC-MS workflow](#) or [LC-IMS-IMS workflow](#)

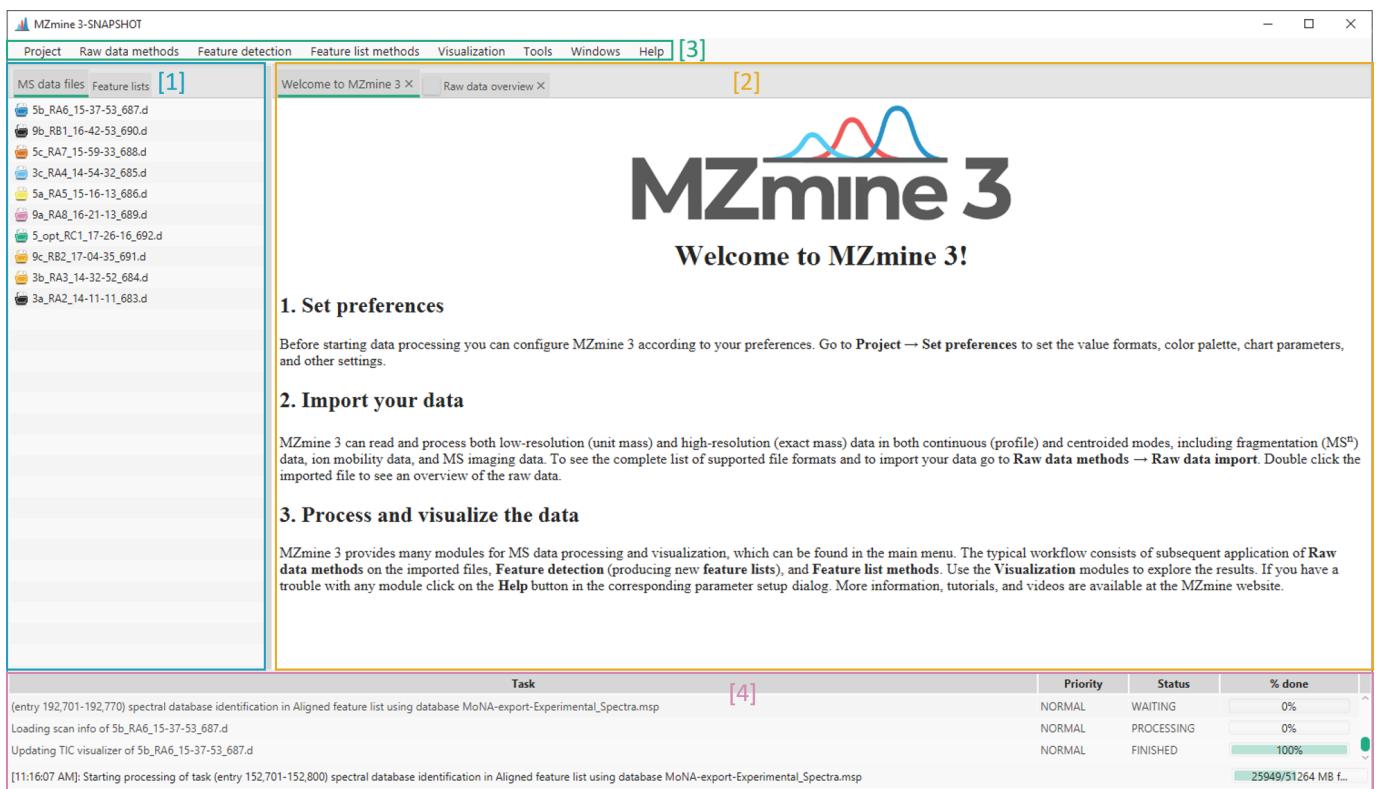
You can also check out the new processing wizard under *Processing wizard* in the main menu.

---

Last update: April 14, 2022 08:25:43

### 3. Main window overview

The MZmine 3 main window is made up of mainly four important building blocks.



#### 3.1 MS data files and feature lists tab

[1]: The (raw) ms data and feature list tabs. Here you can find your imported data files and processed feature lists. *Hint: you can also import files by dragging & dropping them to the ms data tab.*

#### 3.2 Main content pane

[2]: The main content pane. Visualisations such as a raw data overview or a feature list can be viewed here. This pane can also contain multiple tabs. Every tab can also be opened in a new separate window by right-clicking on the header.

#### 3.3 Main menu

[3]: The main menu. Here you can find methods to import and process your data files and feature lists and visualise the results. Furthermore, projects can be saved and preferences can be set.

#### 3.4 Task overview

[4]: The task overview. Current tasks are displayed and their status and progress are indicated. Tasks can also be canceled by right clicking on a task.

## 3.5 Page Contributors

---

[SteffenHeu](#)

---

Last update: April 5, 2022 13:22:07

## 4. Visualization modules

---

### 4.1 Visualization modules

MZmine provides a range of interactive visualization tools for analysis of raw and processed data. JFreeChart library is used for majority of plots. Most of the generated plots are interactive.

The following **functions** are available:

- Use the + or - key on the keyboard to zoom in or out.
- Scroll with the mouse to zoom in or out.
- Drag the mouse from left to right to select the area to zoom in.
- Drag the mouse from right to left to zoom out to the default view.
- Single click on the y-axis to auto set the intensity to auto height.
- Double click on the y- or x-axis to reset the zoom to default.

In the right part of the plot there is a toolbar. Its functionality is also included in a pop-up menu, which appears when you make right click on the plot area.

**TODO:** Add video of working with plots

---

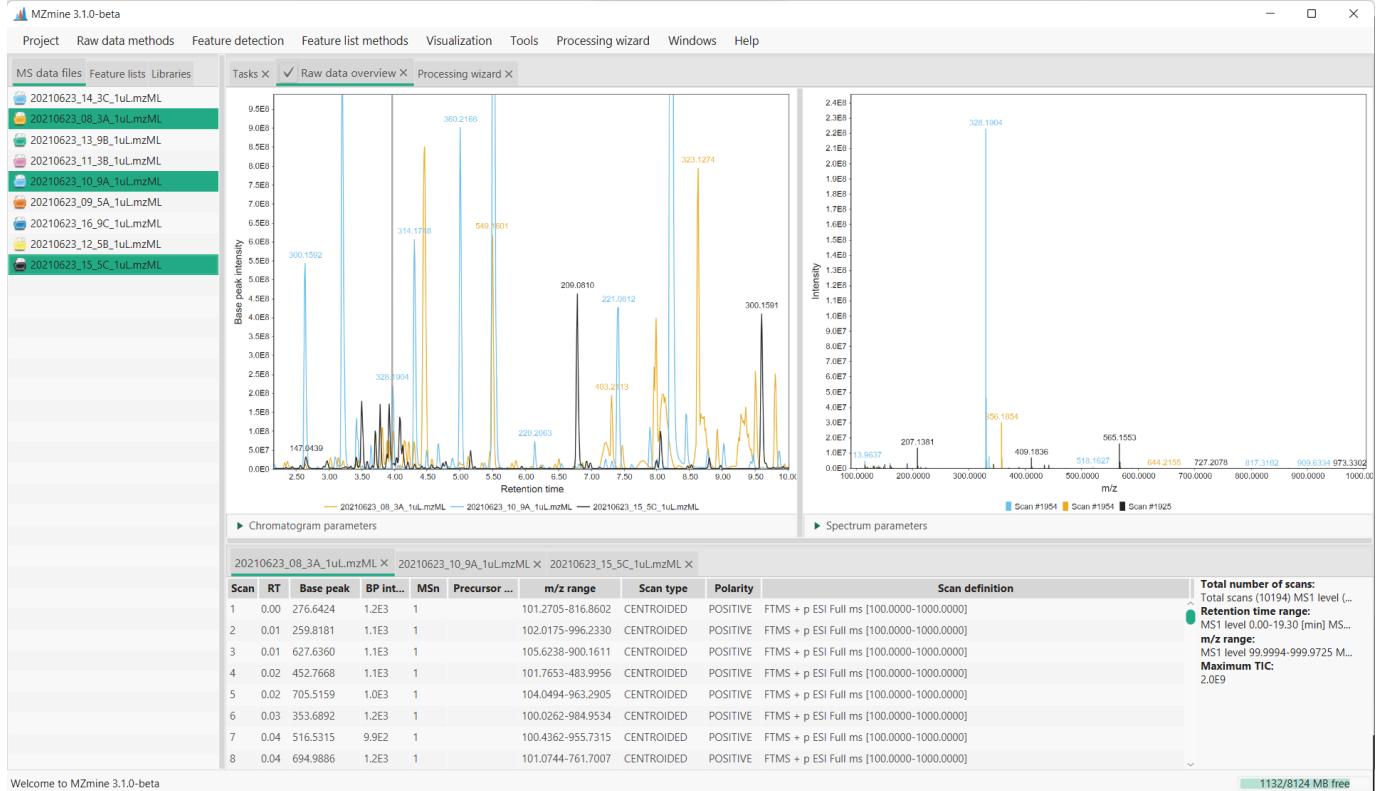
Last update: July 22, 2022 13:14:57

## 4.2 MS data visualisation

### 4.2.1 Raw data overview (LC-MS)

Raw data overview allows user to explore both chromatogram and MS views across all the selected files. If the several files are chosen, they will be displayed in one plot.

The view consists of three panes - chromatogram representation (on the left), mass spectrum (on the right), and table with tabs containing additional information about raw data files.



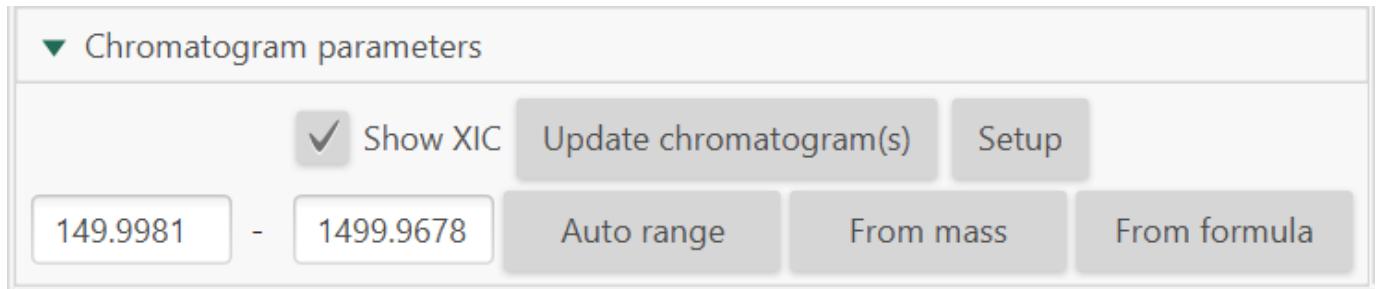
Welcome to MZmine 3.1.0-beta

1132/8124 MB free

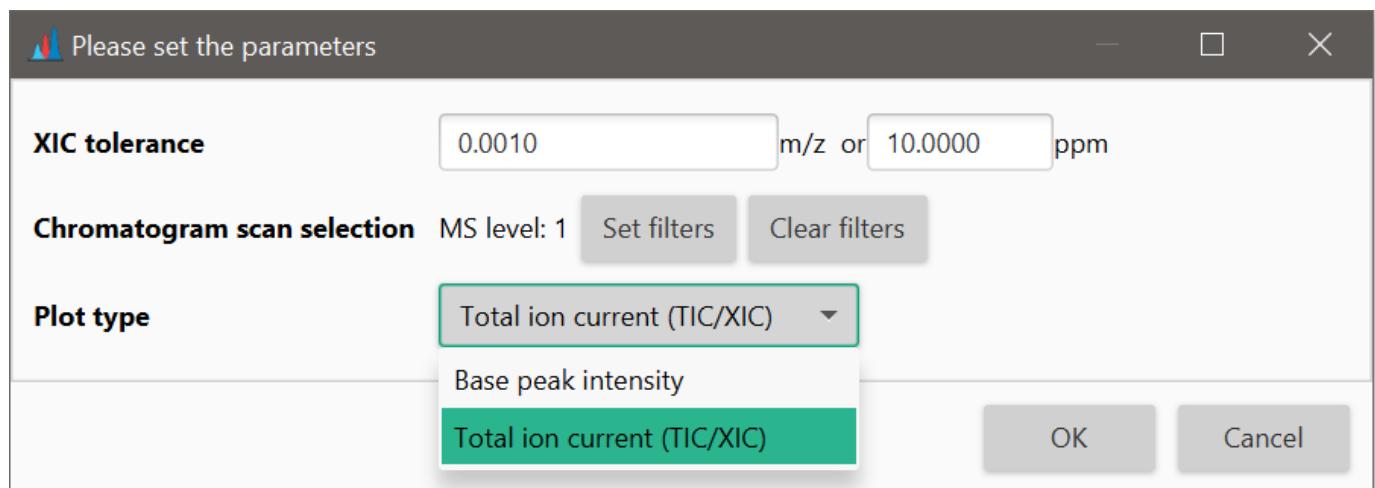
#### CHROMATOGRAM PARAMETERS

##### Show XIC

To display an XIC view of chromatograms in a defined m/z range user can tick an option "Show XIC". m/z range can be either defined manually or generated automatically.



"Setup" button allows to choose appropriate plot type. **Base peak intensity** plot shows the peaks with the greatest intensity while **TIC** plot shows the sum of all the separate ion currents.



#### SPECTRUM PARAMETERS

When masses are detected, it is possible to display them on the spectrum by ticking an option "Show mass list".

#### Chromatogram plot

##### DESCRIPTION

There is a possibility to display chromatographic peaks outside of raw data overview. This standalone two-dimensional plot visualizes TIC/XIC data. All the chromatograms are displayed in the same plot.

The x-axis corresponds to retention time and the y-axis is the intensity level of the signal.

##### PARAMETERS

#### Raw data files

List of raw data files to display in the TIC visualizer.

#### MS level

Scan level (MS1,MS2,... ,MSn) to display in the plot.

#### Plot type

TIC or base peak

#### Retention time

Retention time (x-axis) range.

#### m/z range

Range of m/z values. If this range does not include the entire scan m/z range, the resulting visualizer is XIC type.

#### Selected peaks

List of chromatographic peaks to display in the TIC visualizer. This option is available only if a peak list related to the selected raw data file exists in the current project.

## 4.2.2 MS spectrum

Displays all the ions from a selected scan. Can be used to explore mass spectrum outside of raw data overview. **Only one** raw file can be chosen.

It shows a plot of two dimensions, where X axis corresponds to m/z value and Y axis is the intensity of the ion signal.

**PARAMETERS****Scan number**

Choose the scan to visualize

TODO: Describe MS plot window

**Page Contributors**

Olena Mokshyna

---

Last update: August 1, 2022 19:46:28

### 4.2.3 Additional tools

---

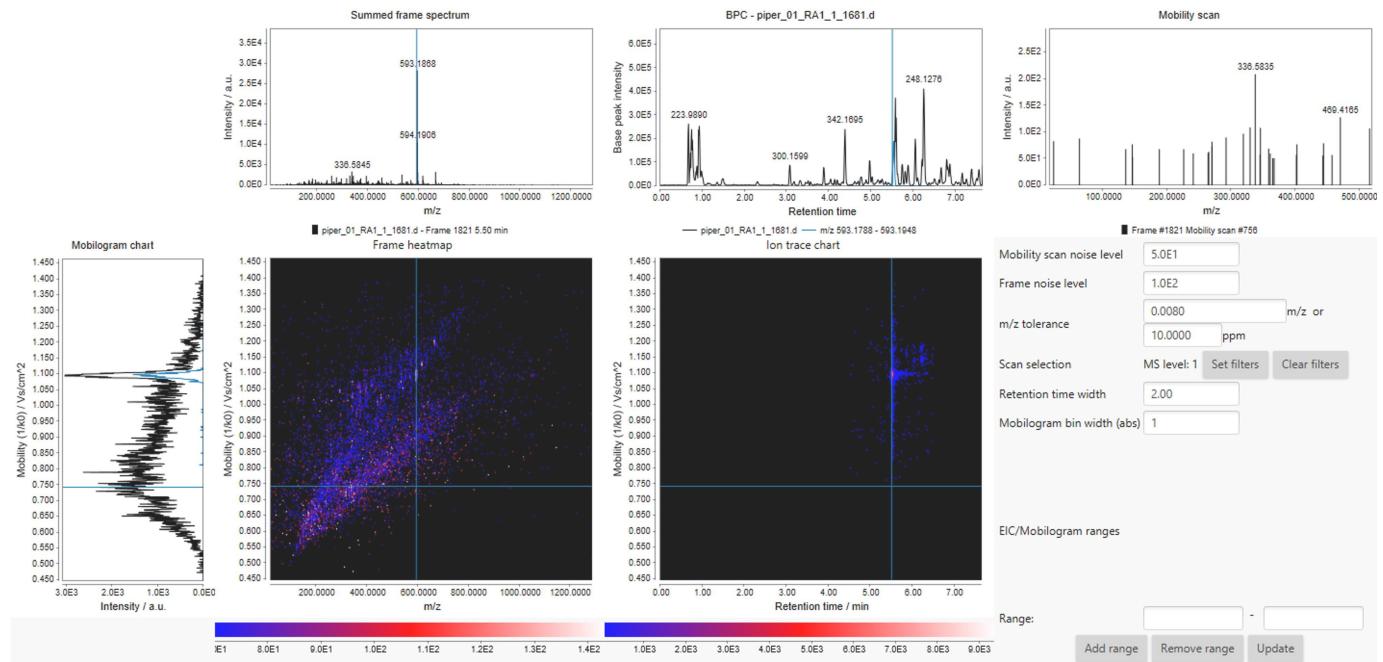
TODO

---

Last update: August 1, 2022 19:46:28

## 4.3 Ion mobility raw data overview (LC-IMS-MS)

The "Ion mobility raw data visualization" module allow a comprehensive navigation of the complex LC-IM-MS raw data. The screenshot below shows an example of LC-IM-MS data acquired with a Bruker timsTOF instrument:



The main window consists of 5 panels and a set of displaying parameters. All the panels are interconnected, which means that moving the cursor in one panel, automatically updates the others. Cursors are displayed as light-blue solid lines in the panels.

### 4.3.1 Summed frame spectrum panel [1]

The MS spectrum corresponding to each **frame** is shown in this panel. The displayed MS spectrum is the sum of all the **mobility scans** acquired over that frame (see [Ion mobility spectrometry terminology](#)).

### 4.3.2 BPC panel [2]

In this panel, the **base peak chromatogram** is displayed. Each data point corresponds to an individual **frame**. Moving the cursor frame-by-frame automatically updates the 'frame heatmap' and 'summed frame spectrum' panels. Moving the cursor frame-by-frame automatically updates the 'summed frame spectrum' panels as changing data point in regular LC-MS data would display a different MS scan. Since each frame is made of several **mobility scans**, the 'mobilogram chart' and 'frame heatmap' panels automatically updates too. *Note*. It is currently not possible to display the [TIC chromatogram](#)

### 4.3.3 Mobility scan [3]

Todo Note that this is the only panel that does not possess a cursor as [...].

### 4.3.4 Mobilogram chart [4]

Todo The signal intensity is displayed as a continuous colour scale.

### 4.3.5 Frame heatmap [5]

Todo The signal intensity is displayed as a continuous colour scale.

---

#### 4.3.6 Ion trace chart [6]

Todo

---

#### 4.3.7 Displaying parameters [6]

**Mobility scan noise level:** This parameter controls the signals shown in the XXX panels (panel n°X). For example, a noise level of 5.0E1 will show only the signals above this value (see below) SCRENSHOT

**Frame noise level:** This parameter sets a threshold for the signals shown in the "Summed frame spectrum panel" (panel n°X). Signals from MS spectra acquired over the same frame are summed and shown

**m/z tolerance:** Todo

**Scan selection:** Todo

**Retention time width:** Todo

**Mobilogram bin width (abs):** Todo

**EIC/mobilogram ranges:** Todo

To-do list: - Explain EIC and EIC in mobilogram chart

---

Last update: March 11, 2022 09:00:25

## 4.4 Image viewer

---

TODO

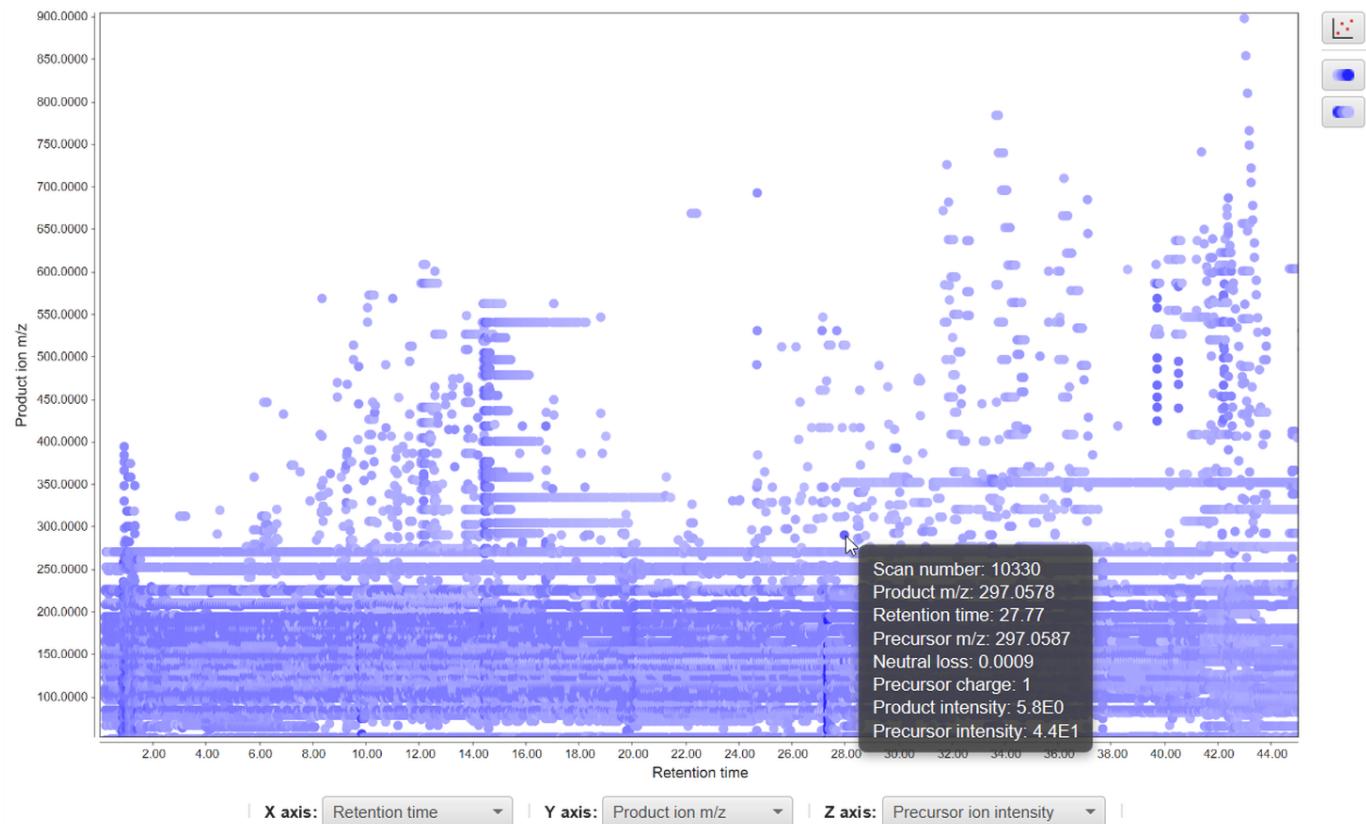
---

Last update: July 26, 2022 12:10:29

## 4.5 Processed data visualition

### 4.5.1 MS/MS plot

This module provides a colored scatter plot of the MS/MS data. There are 4 options for X and Y axes: retention time, precursor ion m/z, product ion m/z, neutral loss and 3 options for Z axis (color): precursor ion intensity, product ion intensity, retention time. The module additionally allows you to filter ions by their intensities and to perform diagnostic fragmentation filtering. In order to focus on the values of interest you can highlight specific data points and sort them by color axis. This tool can be very useful to get an overview of large amounts of MS/MS data by tuning parameters and filters.



#### Parameters

##### Raw data file

Selection of the raw data file to visualize. Only one file can be selected.

##### X axis

Selection of the values for X axis. There are 4 options available: Retention time, Precursor ion m/z, Product ion m/z, Neutral loss.

##### Y axis

Selection of the values for Y axis. Options are the same as for X axis.

##### Z axis

Selection of the values for Z axis. There are 3 options available: Precursor ion intensity, Product ion intensity, Retention time.

##### MS level

MS level of the scans to be plotted.

**Retention time**

Retention time range.

**m/z range**

Range of m/z values for precursor ions in MS<sub>n</sub> scans.

**m/z tolerance**

Maximum allowed difference between two m/z values to be considered same.

**Intensities filtering**

Optional parameter to filter ions by intensity. There are 3 different ways of filtering:

- Number of best fragments - Number of ions with highest intensities from each scan to be visualized.  
For example 5(for each scan 5 ions with highest intensities will be plotted).
- Base peak percent, % - Ions with intensity values lower than the given percent of base peak intensity will be plotted.  
For example 95(ions with intensity values lower than 0.95 multiplied by base peak intensity will not be plotted).
- Intensity threshold - Ions having intensities lower than the given value will not be plotted.  
For example 6.0E6(ions with intensity values lower than 6.0E6 will not be plotted).

**Diagnostic fragmentation filtering**

Optional parameter for diagnostic fragmentation filtering described below. It has 2 subparameters: diagnostic product ions and diagnostic neutral loss values. Scans not containing any ion satisfying each input criterion will not be considered for the visualization.

**Diagnostic fragmentation filtering**

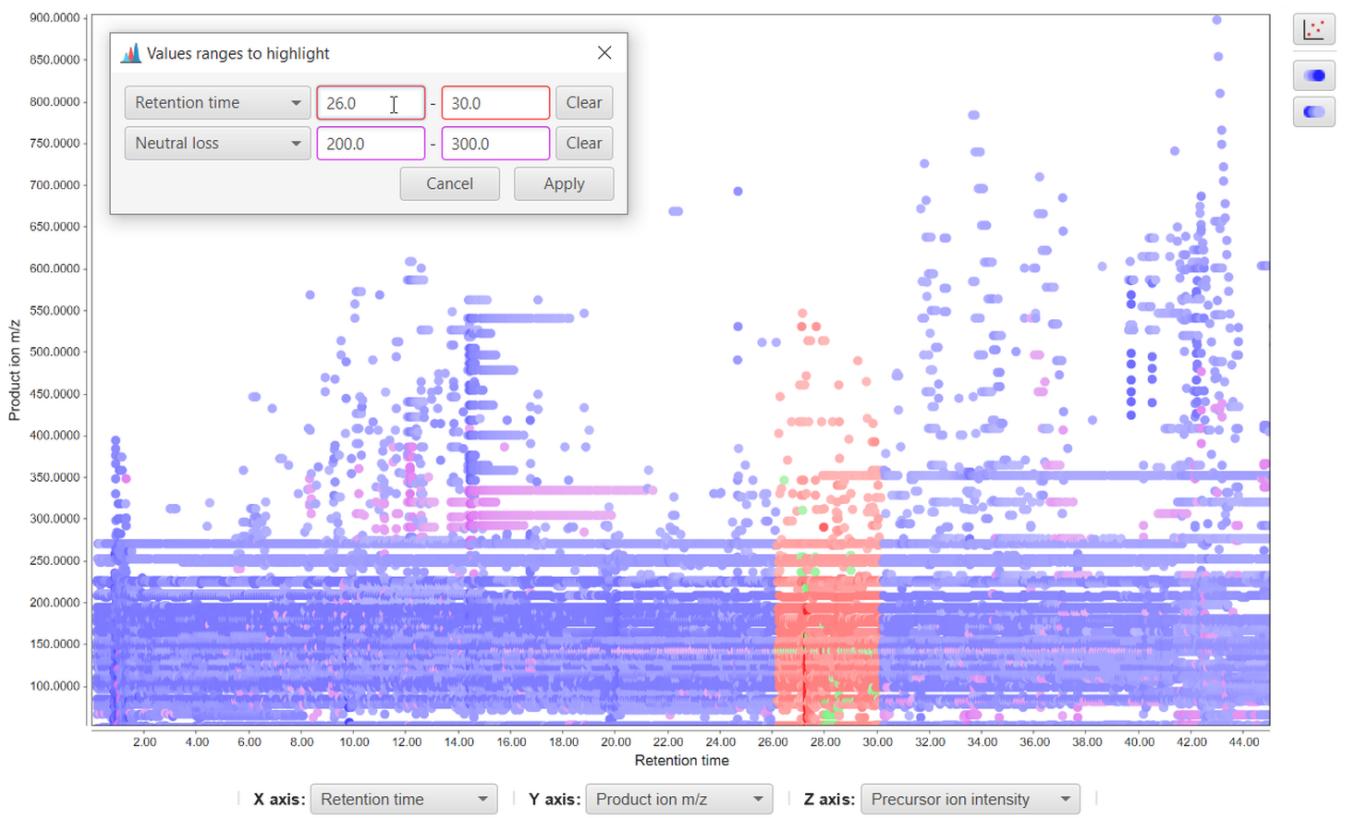
Due to common structural features, compounds within the same class undergo similar MS/MS fragmentation and as a result of many identical product ions and/or neutral losses. Diagnostic fragmentation filter (product ion filter) is a post-acquisition approach to screen LC-MS/MS datasets for entire classes of both known and unknown natural products. This tool searches all MS/MS spectra for product ions and/or neutral losses that has defined as being diagnostic for the entire class of compounds. In other words it screens LC-MS/MS datasets for MS/MS spectra containing production ions and/or neutral losses that are specific to that class of compounds. The user defines the diagnostic product ions and/or the diagnostic neutral loss values (Da) to use in the filtering.

The user can also define the minimum diagnostic ion intensity (% base peak) to use in the filtering. If a recurrent neutral loss occurs, a line pattern in the plot can be observed. If compounds carrying those diagnostic product ions and/or the neutral loss values are detected the resulting plot will show their product ion m/z and precursor ion m/z. Additionally, an output file may be specified that will output the results of the filtering. For a detailed view of diagnostic fragmentation filtering: [Walsh, Jacob P., et al. "Diagnostic Fragmentation Filtering for the Discovery of New Chaetoglobosins and Cytochalasins." Rapid Communications in Mass Spectrometry \(2018\).](#)

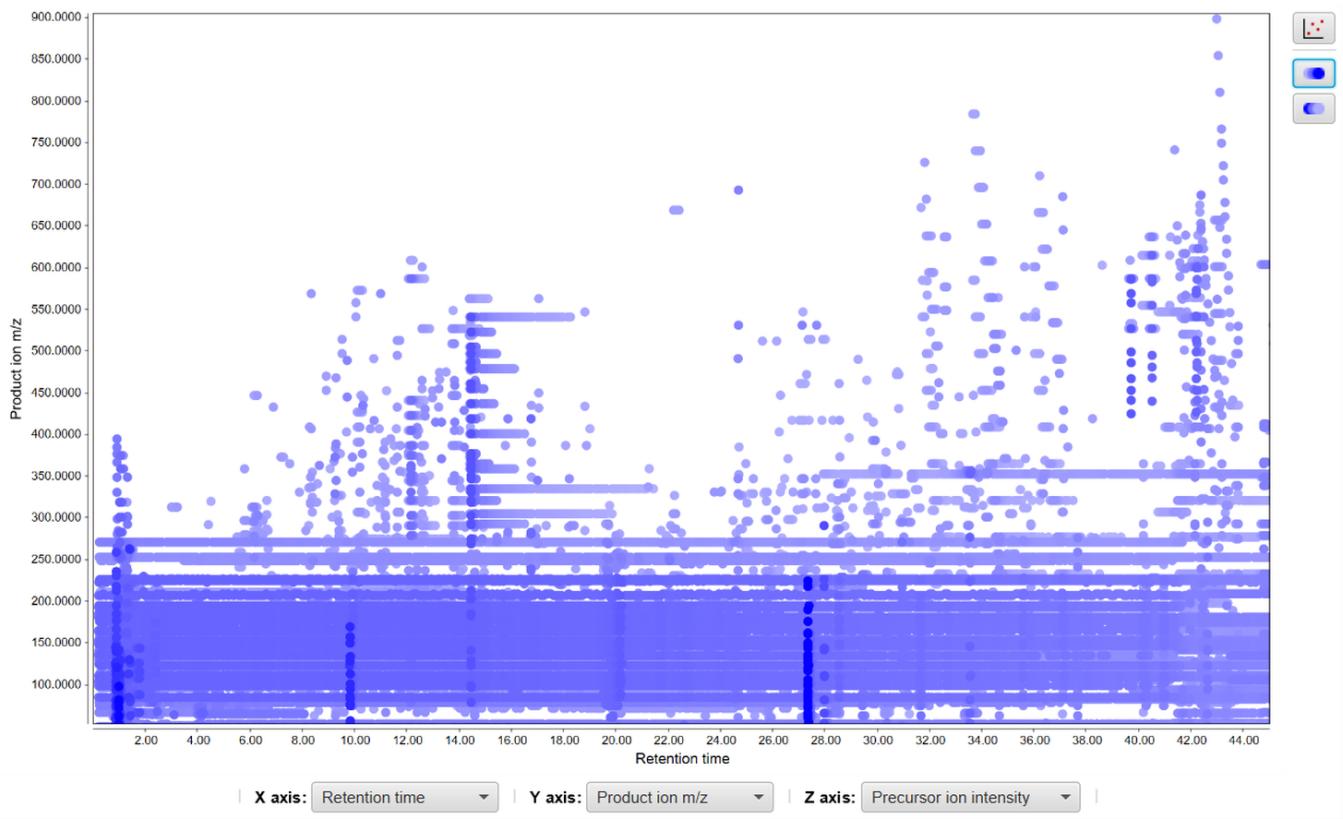
**Functionality**

This plot is using the third part library JfreeChart for its basic functionality.

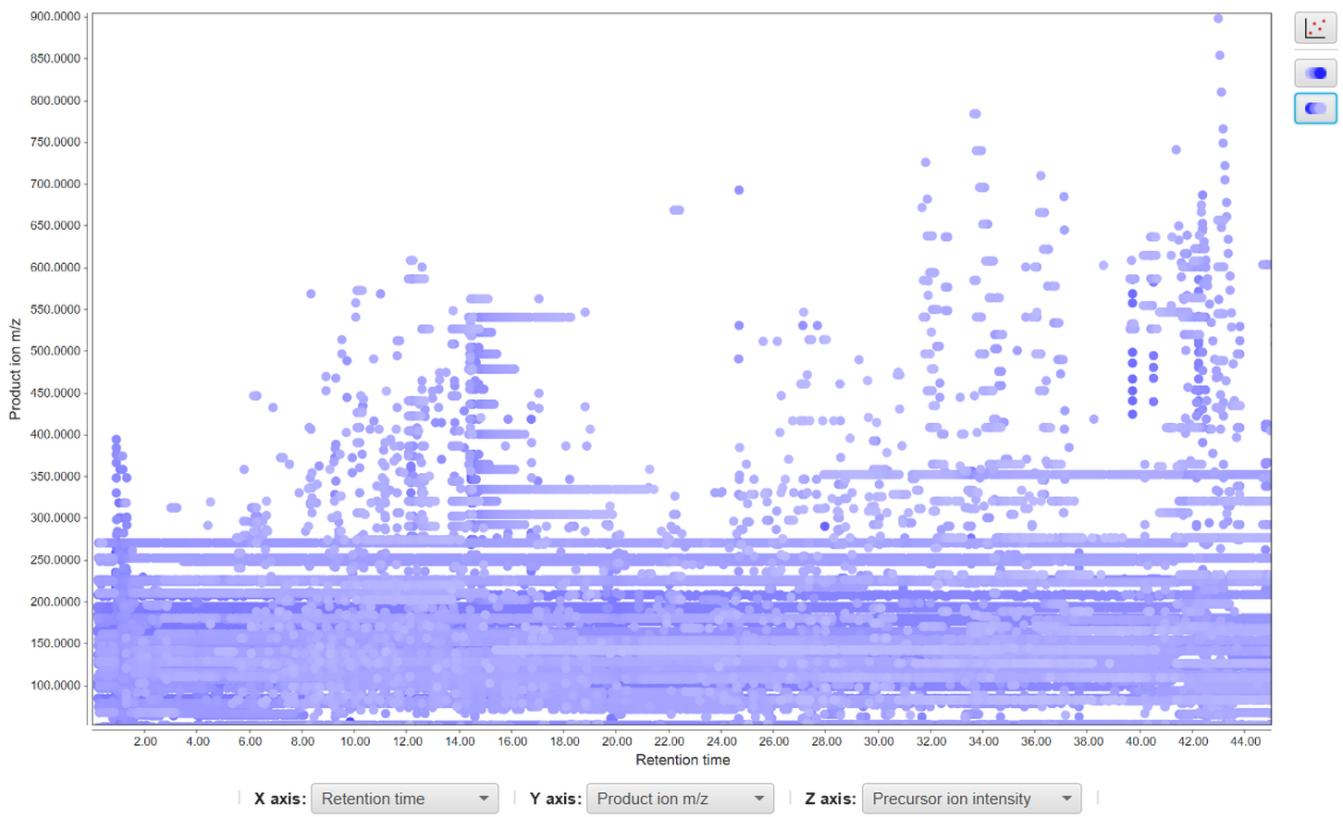
- Drag the mouse from left to right - selecting the area to zoom
- Drag the mouse from right to left - zoom out
- Select combo boxes below - change axes types
- Hold the mouse on data point - show detailed information in a tooltip
- Double click on data point - show spectrum plot
- - highlight points representing ions with specific values given by input ranges (Note: colors of range input boxes determine the highlighting color, green color denotes ions satisfying both ranges)



- - show intense points in front



- show pale points in front



Last update: August 1, 2022 19:46:28

## 4.5.2 Interactive Ion Identity Molecular Networks

---

TODO

---

Last update: July 26, 2022 12:10:29

### 4.5.3 Feature histograms

---

TODO

---

Last update: August 1, 2022 19:46:28

#### 4.5.4 Additional tools

---

TODO

---

Last update: August 1, 2022 19:46:28

## 5. Processing modules

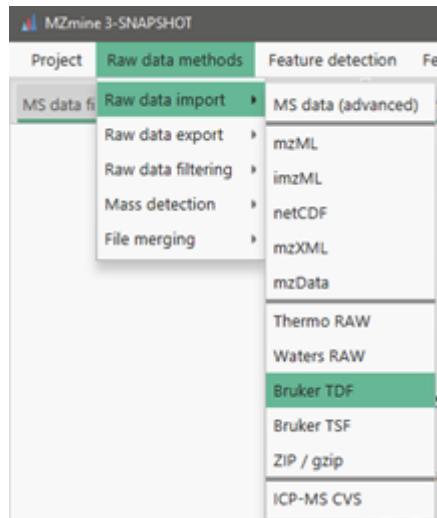
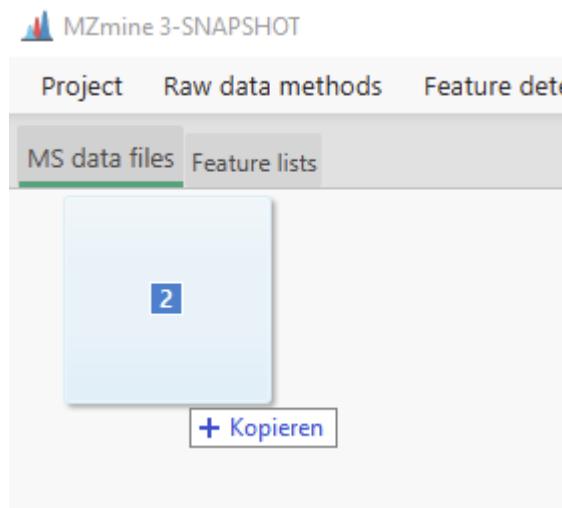
---

### 5.1 Data import

#### 5.1.1 LC-MS data

Raw data can be imported via the main menu **Raw data methods → Raw data import**. Note that multiple data files/folders can be dropped into the **MS data (advanced)** dialog. If individual modules are used, folder based formats can only be imported as one folder at a time. When using the **MS data (advanced)** dialog, inexperienced users should deactivate the direct mass detection steps, since they alter the raw data on the import. Mass detection is then performed, when the scans are loaded and only peaks above the noise level are imported.

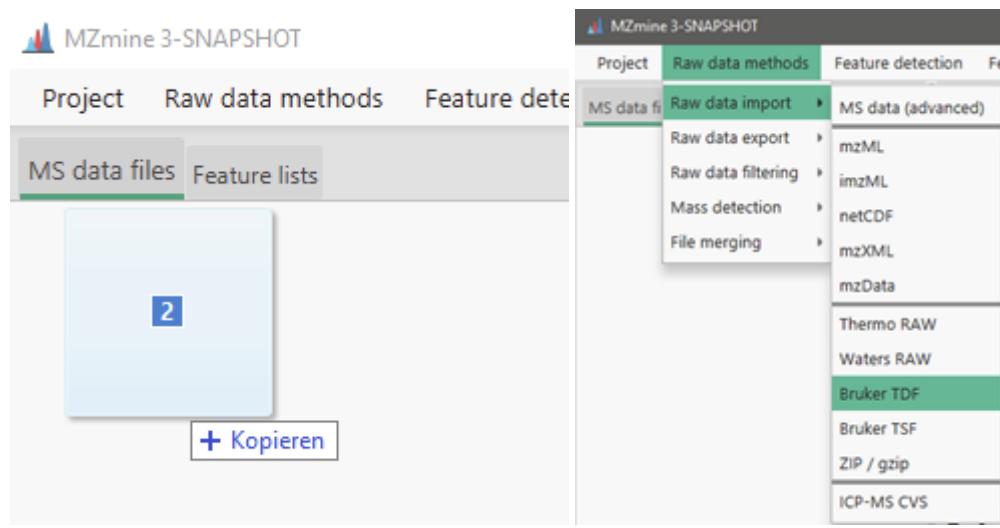
Alternatively, you can simply drag & drop the raw data into the raw data list of the main window.



#### 5.1.2 LC-IMS-MS data

As any other data format, ion mobility data can be imported via the main menu **Raw data methods → Raw data import**. Note that multiple .tdf data folders can be dropped into the **MS data (advanced)** dialog. The Bruker TDF import can only select a single folder. When using the **MS data (advanced)** dialog, inexperienced users should deactivate the direct mass detection steps, since they alter the raw data on the import. Mass detection is then performed, when the scans are loaded and only peaks above the noise level are imported.

Alternatively, you can simply drag & drop the raw data into the raw data list of the main window.



Last update: March 10, 2022 15:53:40

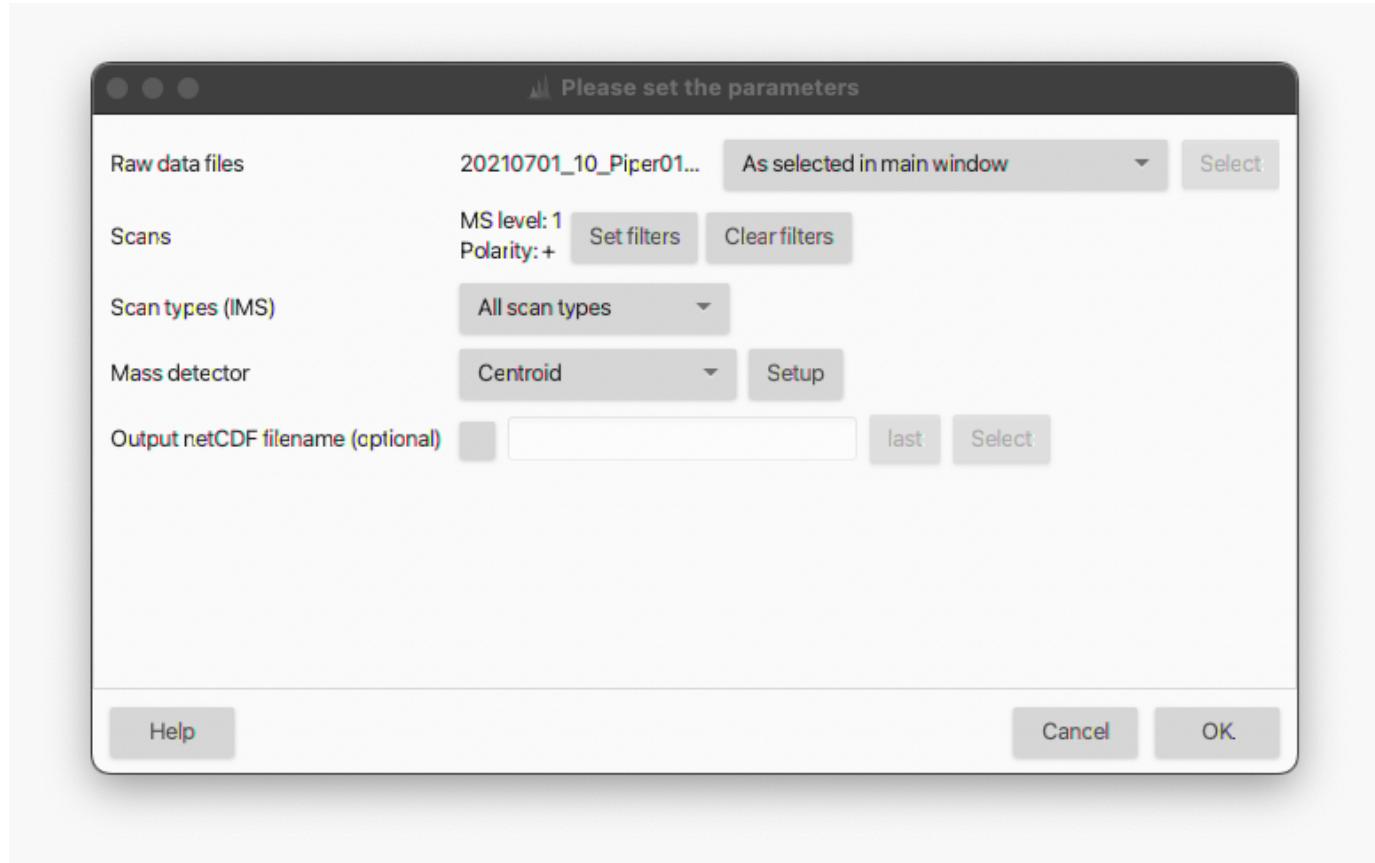
## 5.2 Mass detection

### 5.2.1 Mass detection

The mass detection module generates a [mass list](#) (*i.e.* list of m/z values and corresponding signal intensities) for each scan, in each raw data file. During the mass detection, profile raw data are centroided and a noise filtering is performed based on a user-defined threshold (see [Setting the noise level](#)).

#### Parameters settings

:material-menu-open: Raw data methods → Mass detection → Mass detection



#### Raw data files

Select the input raw data files for the mass detection. All the imported data files can be processed in bulk (*i.e.* *All raw data files*), or a subset can be selected directly from the *MS data files* panel (*i.e.* *As selected in the main window*) or based on the filename (*i.e.* *File name pattern*). As an alternative, the files' directory can be also specified (*i.e.* *Specific raw data files*). Finally, if the *Those created by previous batch step* option is selected, MZmine takes the output of the last processing step as input. This option is only available for [batch processing](#).

#### Scans

Select (or filter out) the MS scans to be processed. Several filters are available (*Select filters* button). A scan number, RT and mobility range can be set (*i.e.* *Scan number*, *Retention time* and *Mobility* options); only the scans belonging to the defined range(s) will be processed. The *Base Filtering Integer* option allows to process one every-N scans. The *Scan definition* field can be used to filter scans based on the scan's description normally included in the raw file's metadata (*e.g.* FTMS). Scans can also be filtered by *MS level* (*i.e.* 1, 2, ..., N), polarity and spectrum type (*i.e.* Centroided, Profile and Thresholded).

### Scan types (IMS)

This parameter applies only to IM data and determines if *mobility scans*, *frame scans* or both (*i.e.* All scan types) are processed. For more details about *mobility* and *frame scans*, see [here](#).

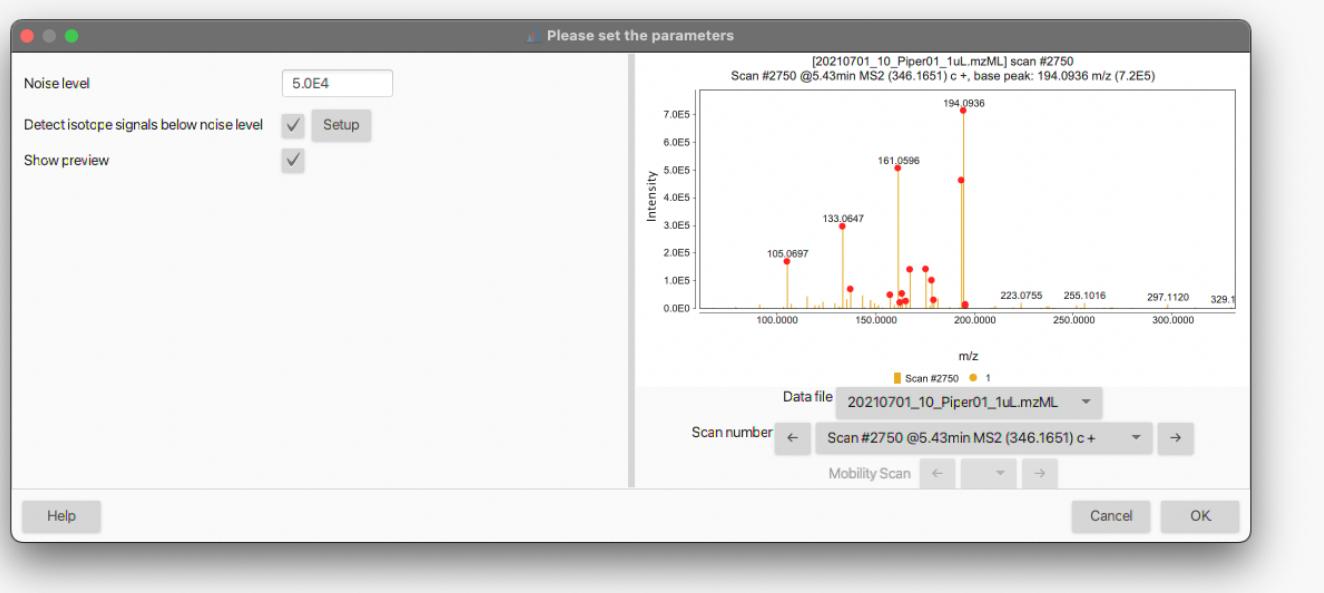
:octicons-light-bulb-16: **Tip.** Since *frame scans* are obtained by merging multiple *mobility scans*, the noise thresholds will likely be different. However, only one noise level can be set per processing step. Therefore, if one wants to run the mass detection for *mobility* and *frame scans* using different noise cutoffs, two module calls are required. As an alternative, mass detection can be performed only on the *mobility scans* by selecting the appropriate noise level. *Mobility scans* can then be merged into *frame scans* with a [dedicated module](#).

### Mass detector

Select the algorithm to be used for the mass detection. Several mass detection algorithms are available and can be selected in the drop-down menu. The choice depends on the raw data characteristics (profile/centrotoed, mass resolution, etc.). The *Centroid* algorithm must be used for already-centrotoed data. A step-by-step guide to convert profile into centrotoed data is provided in the [GNPS documentation](#). Other algorithms are available for profile raw data and are described in more details [here](#). The *Exact mass* algorithm is highly recommended for profile HRMS data. When *Auto* is selected, the *Centroid* and *Exact mass* algorithms are used by default for centrotoed and profile data, respectively.

### SETTING THE NOISE LEVEL

All the mass detection algorithms allow to set a threshold for the noise filtering (*i.e.* *Noise level*) by hitting the *Setup* button next to the *Mass detector* field. A dialog box like the following will open up:



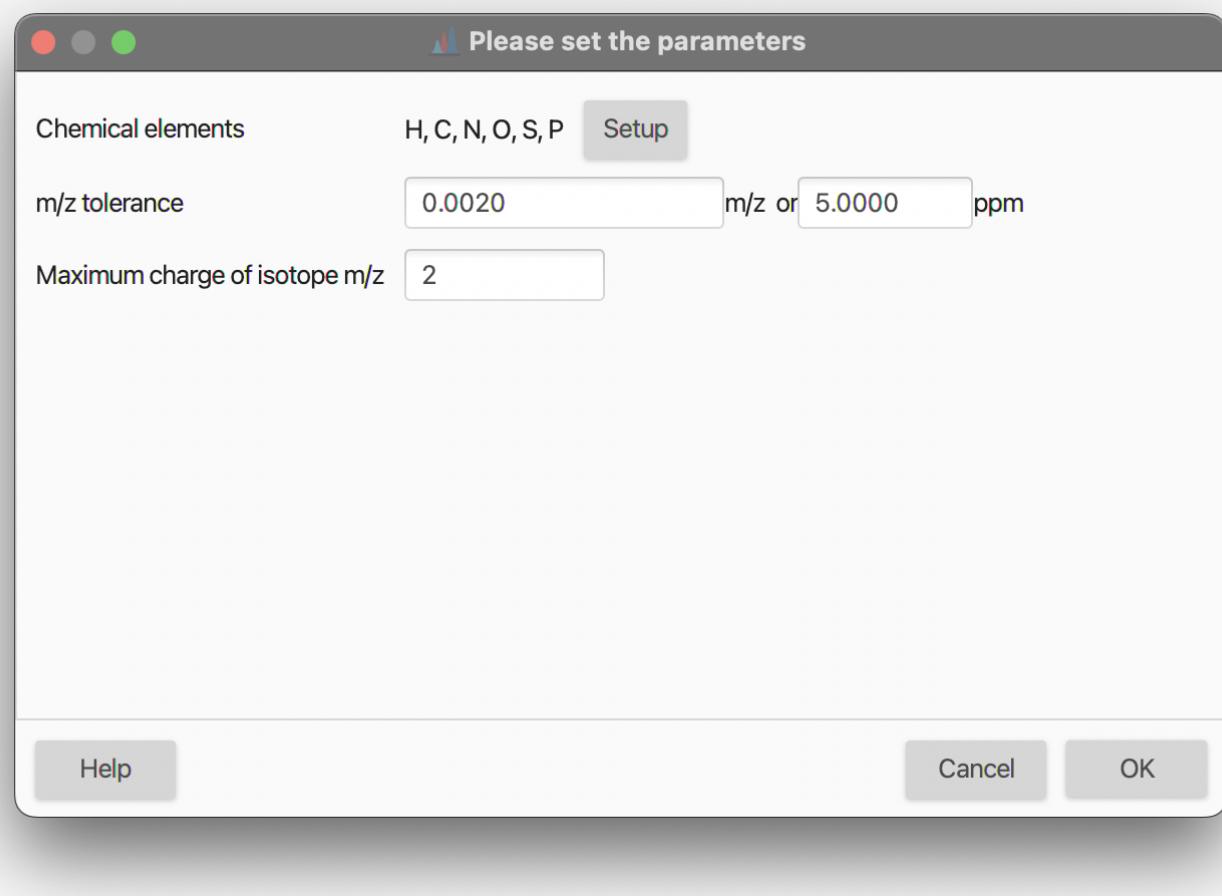
The noise threshold can be entered either in standard or scientific notation. By checking the *Show preview* box, an interactive visualization panel will open to help the user to adjust the noise level (see also [How do I determine the noise level in my data?](#)). The red dots denote the mass signals retained in the mass list according to the set threshold. Different data files and scan numbers can be visualized using the corresponding drop-down menus.

### DETECT ISOTOPE SIGNALS BELOW NOISE LEVEL

The *Centroid* and *Exact mass* algorithms provide the option to retain signals that are below the noise level (and would be otherwise discarded), but correspond to isotopes of the detected masses. Theoretical isotopic distributions are calculated for each mass detected in the *mass list* based on the specified chemical elements. If a signal below the noise threshold that matches a theoretical isotopic mass is found in the raw data, it will be included in the final mass list.

:octicons-light-bulb-16: **Tip.** In the case of LC-MS data processing, the low-intensity isotope signals included in the final mass list will undergo the whole feature detection workflow (see, for example, [LC-MS data processing workflow](#)). Due to the low intensity, these masses often produce LC peaks with poor peak shape during the chromatogram building step and might be discarded if they do not meet the user-defined parameters (*e.g.* minimum number of data points and intensity, see [ADAP chromatogram builder](#) for more details). Therefore, it might be advisable not to use this option during the mass detection, but rather use the Isotope finder module (CREATE DOC).

By ticking the corresponding checkbox and hitting the *Setup* button, the following dialog box opens up:



#### Chemical elements

Elements considered when generating the isotopic distributions. Select the elements from the periodic table by hitting the *Setup* button.

#### m/z tolerance

Maximum allowed difference between measured and theoretical isotope  $m/z$ . It is an [intra-scan m/z tolerance](#). The tolerance can be set in  $m/z$ , ppm or both. Since mass deviations expressed in ppm are dependent on the  $m/z$  (*e.g.* higher at low  $m/z$  and lower at high  $m/z$ ), MZmine automatically uses the largest tolerance.

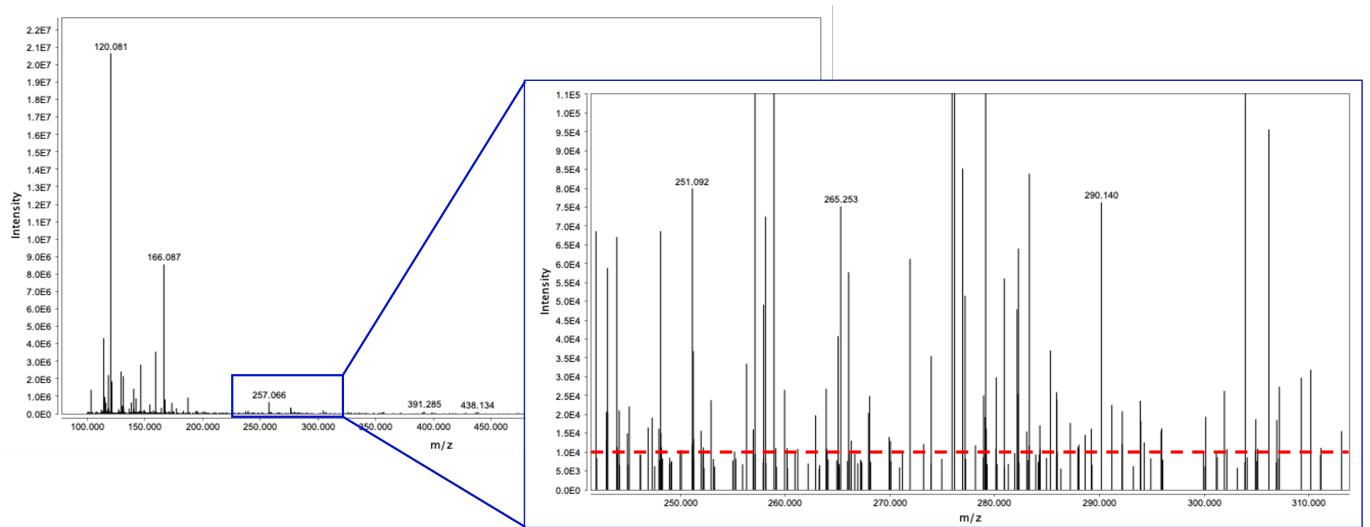
#### Maximum charge of isotope m/z

Maximum allowed charge state of the isotope to be retained in the mass list. Default value is 1.

#### HOW DO I DETERMINE THE NOISE LEVEL IN MY DATA?

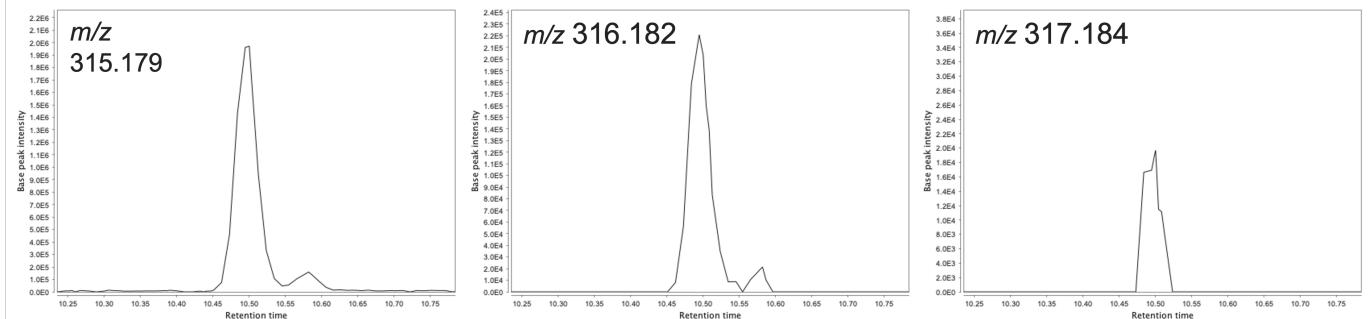
The background noise level largely depends on the mass spectrometer and detector type. For example, Orbitrap instruments normally provides higher signal intensities than TOF devices. To provide some numbers, while 1.0E2 - 1.0E3 could be an appropriate noise level for TOF analyzers, the same would be overly low for Orbitrap instruments (which normally require 1.0E4 - 1.0E5).

The best way to determine the instrumental noise level is undoubtedly by looking at the raw data. The background noise (often referred to as "grass" in technical jargon) is characterized by several signals having the same intensity and no clear pattern among them (see Figure).



Since these signals are produced by electrical and/or mechanical noise, rather than actual ions being detected, they should be excluded from the mass detection and downstream data processing. The red dashed line in the figure corresponds to a hypothetical noise level (1.0E4 in this case) that would filter out most of the "grass"-type noise from the mass detection.

Another way, more relevant for the feature detection, to determine the noise level consists of picking a mid-intensity LC peak and extract the EICs of its  $^{13}\text{C}$  isotopes. When the chromatographic peak shape starts to deteriorate, it means we are approaching the instrument detection limit (see Figure).



Such approach can also be useful to determine other parameters in the feature detection such as the [Group intensity threshold](#) and [Min highest intensity](#) parameters in the [ADAP chromatogram builder](#) module.

Last update: May 11, 2022 09:54:27

## 5.2.2 Mass detection algorithms

Mass detection can be done with the following six algorithms:

- Centroid
- Factor of the lowest signal
- Exact mass
- Local maxima
- Recursive threshold
- Wavelet transform
- Auto

### Centroid

Centroid algorithm assumes that each signal above a given noise level is a detected ion. This mass detector is suitable for already centred data.

### Factor of the lowest signal

Description

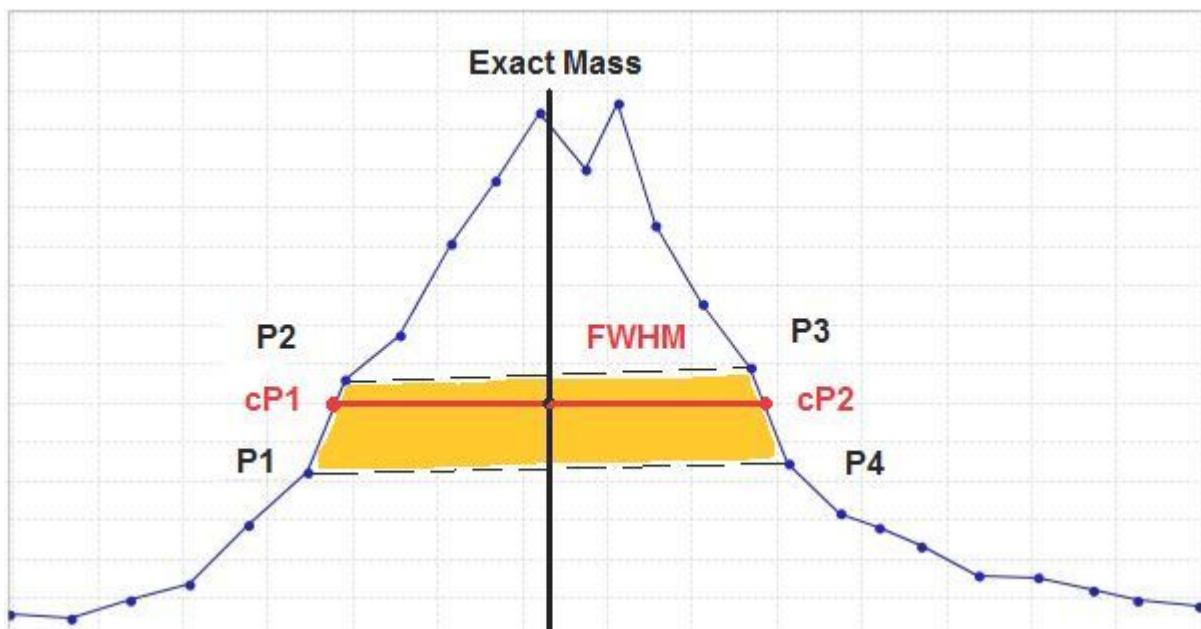
This mass detector is suitable for already centred data. It removes all data points below a spectrum's lowest intensity multiplied by a factor.

### Exact mass

This mass detector first searches for all local maxima within the spectrum, which then form candidate ions.

This method calculates the exact mass of a peak using the **FWHM** (full width at half maximum) concept and linear equation ( $y = mx + b$ ). FWHM is the difference between the two values of the independent variable at which the dependent variable equals half of its maximum value.

First, the method locates the data points located nearest to the peak center at half of the maximum intensity (P1, P2, P3, P4). With these four points it calculates two points (cP1, cP2) that define the width of the peak. The exact mass is then obtained as the center of the width.



This method is suitable for high-resolution MS data, such as provided by FTMS instruments.

**Tip:** The exact mass algorithm is highly recommended for profile MS data.

**Tip** If the continuous data is too noisy, one can use recursive threshold algorithm.

#### Local maxima

This very simple mass detector detects all local maxima within the spectrum, except the signals below the given noise level. The practical usability of this method on real MS data is limited, but it is useful to demonstrate and understand the functionality of mass detection using the preview plot.

#### Recursive threshold

The algorithm finds all m/z ranges within the given limit in a recursive way.

Initially, it looks at the whole range of data points. If the m/z width of this range is not within given limits, a minimum data point is found and used to split the range in two parts. The same algorithm is then applied recursively on each part. Recursion continues until all m/z ranges fitting into the given width limits are found.

Final m/z values are determined as local maxima of the identified m/z ranges.

This mass detector is suitable for continuous data, which has too much noise for the Exact mass detector to be used, but which shows a consistent width of m/z peaks.

#### Additional method parameters

**Min m/z peak width:** Minimum acceptable peak width in m/z.

**Max m/z peak width:** Maximum acceptable peak width in m/z.

**Tip** Recursive threshold method can be used with the noisy continuous data that shows a consistent width of m/z peaks.

#### Wavelet transform

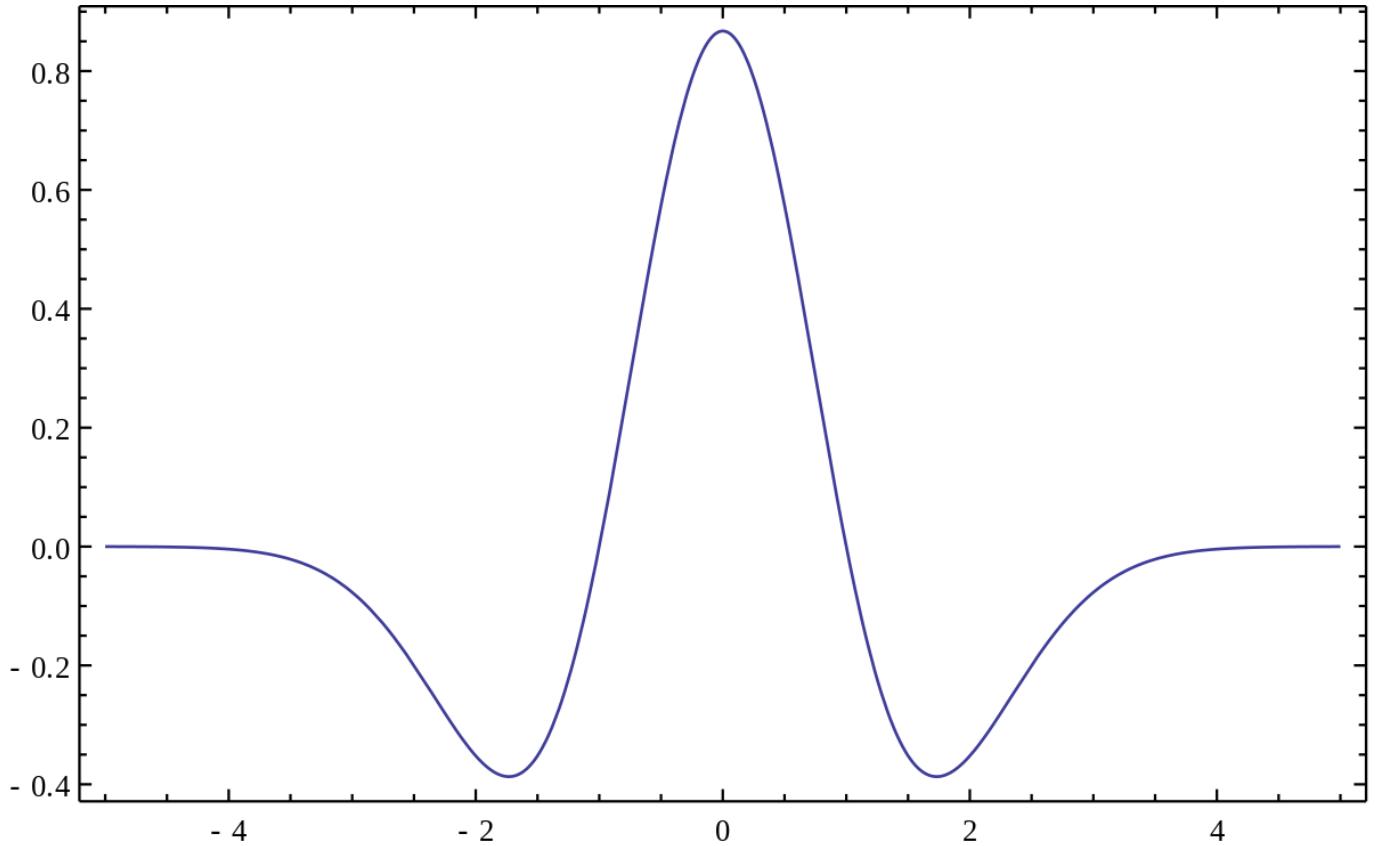
The method uses the **Mexican Hat wavelet model** of the continuous wavelet transform (**CWT**) algorithm.

The search of mass spectrum peaks is executed in three steps. First, the data point intensity is converted into wavelet domain. Second, all the local maxima of the calculated wavelet are found. Finally, m/z peaks (ions) are declared in those points, where the wavelet has a local maximum. The m/z peak is formed with the selected data point (mass and intensity) using the wavelet and all surrounding data points. The final m/z value of the ion is calculated as an average of m/z values of surrounding data points weighted by their intensity.

#### Mathematical model

In numerical analysis, the Mexican hat wavelet is the normalized second derivative of a Gaussian function.

$$\psi(t) = \frac{1}{\sqrt{2\pi}\sigma^3} \left( 1 - \frac{t^2}{\sigma^2} \right) e^{-\frac{t^2}{2\sigma^2}}$$

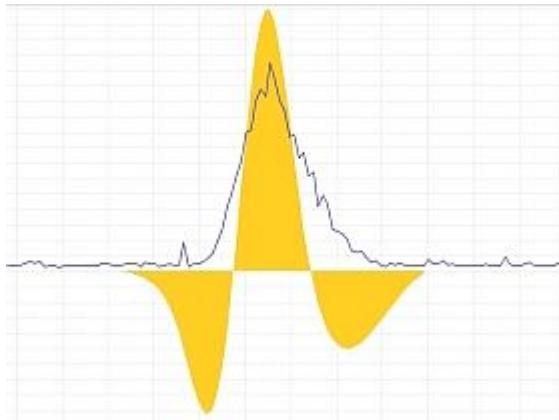


The parameter  $\langle t \rangle$  is the intensity of each data point in the curve, and  $\langle \sigma \rangle$  corresponds to the standard deviation.

To simplify the process of wavelet calculation, the original function is transformed into two parts, where  $\langle W_c \rangle$  is the wavelet coefficient and  $\langle y \rangle$  is the intensity of the wavelet at certain point. In the following formula,  $\langle t \rangle$  corresponds to the Wavelet window size (%) parameter.

```

$$\langle W_c \rangle = \left\langle \frac{2}{\sqrt{3}} \pi^{\frac{1}{4}} \right\rangle \left( 1-t^2 \right) e^{-\frac{t^2}{2}}$$
 $ y = \lim_{LL \rightarrow UL} \langle W_c \rangle x \right]
```



The lower (LL) and upper (UL) limits, where the Mexican Hat wavelet is evaluated, are from -5 until 5. The incremental step used in this range is the result of limits range division by 60,000.

Additional method parameters

### Scale level

Number of wavelet coefficients to use in m/z feature detection. Serves as the scale factor that either dilates or compresses the wavelet signal.

When the scale factor is relatively low, the signal is more contracted, which results in a more detailed resulting graph and more noisy peaks are detected. On the other hand, when the scale factor is high, the signal is stretched out, which means that the resulting graph will be less detailed with a smoothed signal.

**Wavelet window size (%)** The size of the window used to calculate the wavelet signal. When the size of the window is small, more noisy peaks can be detected. The proper value of this parameter may help to avoid the undesired noise peaks.

 **Tip** The Wavelet transform mass detector is particularly suitable for low-resolution and noisy data.

#### Auto

Auto mass detector recognizes if the spectrum is of profile or centroided data type and applies centroid or exact mass algorithms correspondingly.

---

Last update: August 1, 2022 17:24:00

### 5.2.3 FTMS shoulder peak filter

#### Description

**!** This module should be used after mass detection step is performed.

Raw data obtained from FTMS (Fourier Transform Mass Spectrometer) instruments often contains false signals around high-intensity m/z peaks, called "**shoulder peaks**". These signals are residues of the Fourier Transform function and their intensity is usually below 5% of the main (true) m/z peak.

The FTMS shoulder peaks filter attempts to remove these false signals. Ions in the mass lists (generated previously by the Mass detector module) are processed in the order of decreasing intensity. A peak model (shape) is built around each ion peak using given function and resolution, and those m/z peaks which fall below the model are considered to be shoulder peaks and therefore are removed.

The method offers three theoretical peak models.

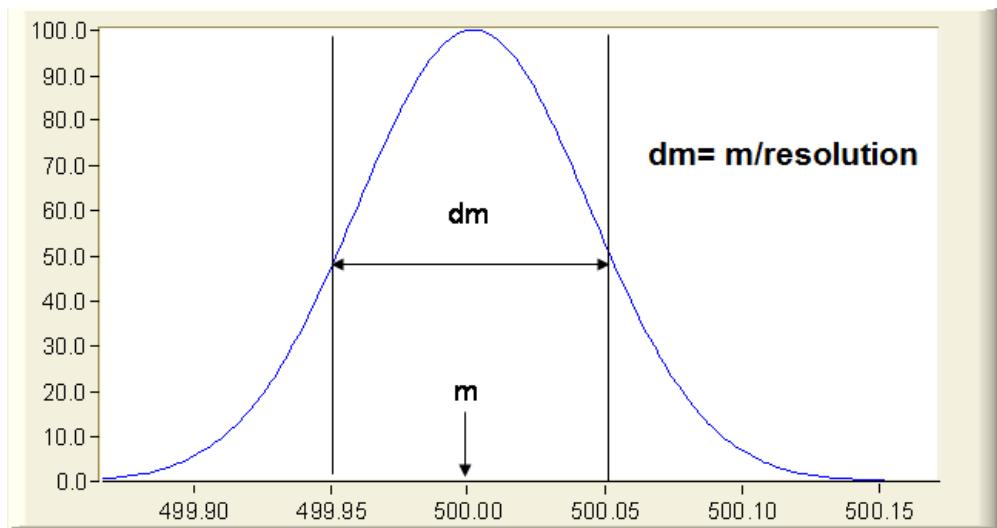
Example of running the shoulder peaks filter on LTQ Orbitrap data:



#### Method parameters

##### Mass resolution of the data

Defines the width of the model, which should be equal to the estimated resolution of the peaks in the raw data. Mass resolution is the dimensionless ratio of the mass of the peak divided by its width. Peak width is taken as the full width at half maximum intensity (FWHM).



### Peak model function

Defines the shape of the model function, as described below. Peaks under the curve of this peak model will be removed.

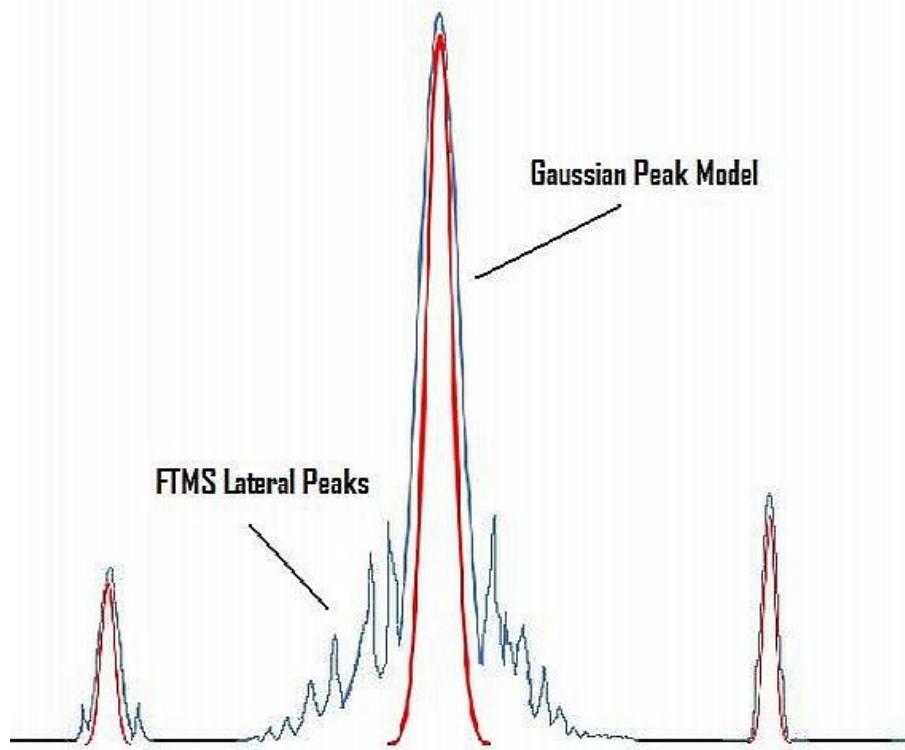
#### Available peak models

##### GAUSSIAN PEAK MODEL

The Gaussian peak model is a characteristic symmetric "bell shape curve" that quickly falls off towards plus/minus infinity, described by the following formula.

$$f(x) = ae^{-\frac{(x-b)^2}{2c^2}}$$

The parameter "a" is the height of the curve's peak, "b" is the position of the center of the peak, and "c" controls the width of the "bell".

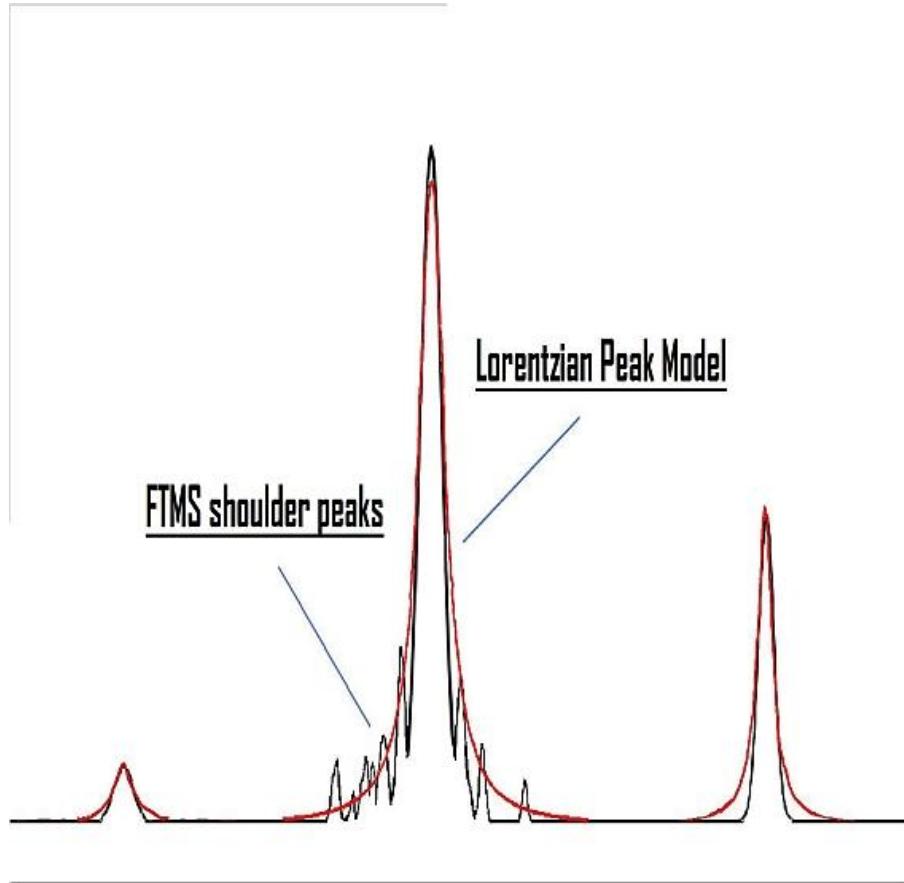


#### LORENTZIAN PEAK MODEL

The Lorentzian function (Cauchy-Lorentz distribution) is used for this model. The Lorentzian peak model is described by the following formula:

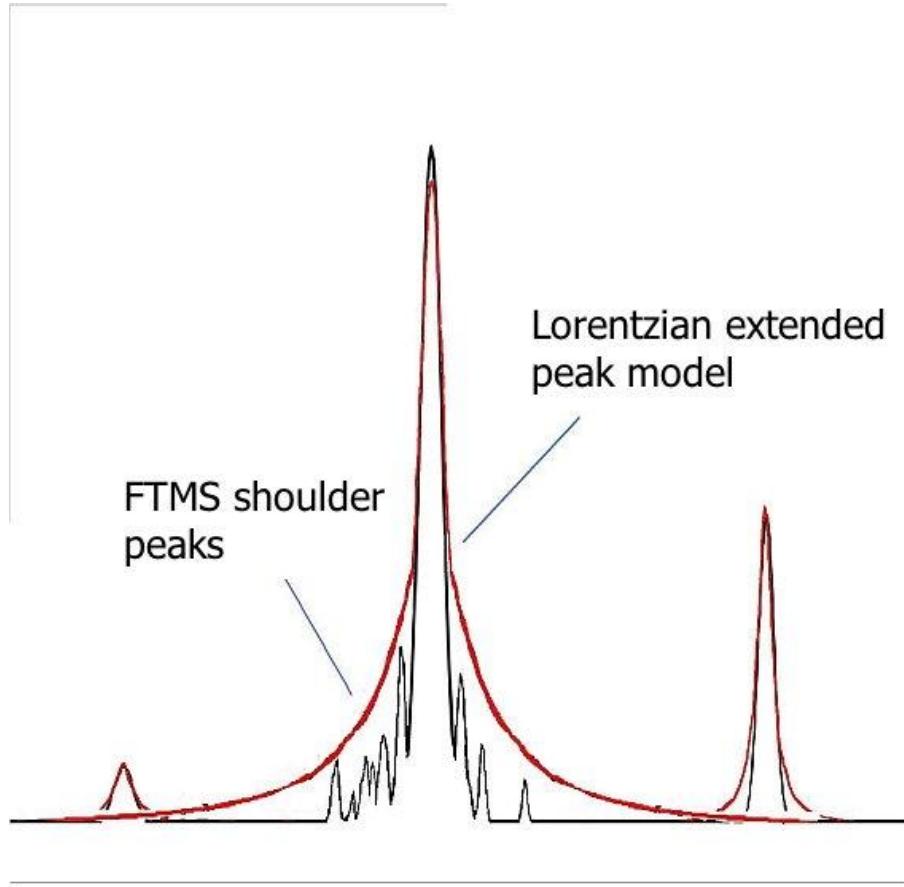
$$f(x; x_0, \gamma, I) = \frac{I}{\gamma} \frac{\gamma^2}{(\gamma^2 + (x - x_0)^2)}$$

Where "x0" is the location parameter, specifying the location of the peak of the distribution, and "y" is the scale parameter which specifies the width of the peak.



#### LORENTZIAN EXTENDED PEAK MODEL

This model uses the same mathematical formula as the Lorentzian peak model, but the lower part of the model (below 5% of the intensity) is extended. The width of the peak below 5% intensity is calculated from another Lorentzian peak with 5% of the resolution of the main peak.



Last update: August 2, 2022 07:55:31

## 5.2.4 Mass calibration

---

### Description

 This module should be used after mass detection step is performed.

Mass calibration module estimates the measurement error of each detected mass and calibrates them using reference libraries of ions through three main processes:

1. Peak matching with library of reference and extraction of errors,
2. Estimation of overall mass bias, and
3. Mass calibration of detected masses.

#### REFERENCE LIBRARY OF IONS

First, mass lists (MS1) from raw data files are matched against a reference library and mass measurement errors are calculated.

This module can support two different matching strategies:

- **Standard Calibrant Library (SCL)** (recommended method): file needs to be provided by the user in xls, xlsx or csv format). The file needs to include retention time (RT) information and the ion formula of a collection of ions that are expected to appear in the samples analyzed with a known chromatographic method. Ion formulas format of the SCL will depend on the ionization method used to analyze the samples. See below in the “Parameters” section a detailed description for the library format.
- Using a SCL, the matching of measured peaks the matching of measured peaks against the library is performed using both RT and exact m/z. RT and m/z error tolerances are needed to be defined.
- :octicons-light-bulb-16: Note: To ignore RT parameter when using SCL for mass calibration, the parameter can be set with an equal or larger value of the chromatography length (i.e., 30 min).

- **Universal Calibrant Library (UCL)** include a collection of ions often found in mass spectrometry experiments. The module has two different lists in positive (+ve) and negative (-ve) ionization modes to be chosen by the user.

Universal calibrant lists are based on **Keller et al. 2008 *Analytica Chimica Acta* 627:71-81** and **Hawkes et al. 2020 *Limnology and Oceanography Methods* 18:235-258**. MZmine includes both those libraries.

Matching of detected ions against the UCL library is performed using m/z values alone independently where the ions are appearing along the chromatography and only m/z error tolerance will be needed.

#### OVERALL MASS BIAS ESTIMATION

Measurement mass errors are calculated based on the matching of detected ions against the library, a distribution of errors is built, and measurement bias is estimated. Because not all ion matches can be considered as correct, the calibration model will automatically identify the high-density mass error range (mass error range with larger number of matches) from the generated distribution of matching errors.

For the overall mass bias estimation, two methods can be used:

- **Percentile Range of Errors:** The module calculates the Interquartile Range (IQR) from the overall distribution of errors to extract those errors to be used for mass calibration of peaks. IQR can be modified (by default determined at 25th and 75th percentiles). In such case, errors distributed below 25th or above 75th percentile will not be considered for mass calibration of peaks.
- **High-Density Range of Errors:** Autonomous method relying on two parameters set by the user. This method is performed relies in two parameters:
  - *Primary High-Density Range of Errors size (in PPM):* finds the highest density range of errors (in PPM). The module will find the distribution range containing most mass errors within the set PPM error (i.e., 2 PPM).
  - *Error range tolerance (in PPM):* determines how far the distribution range of errors will be extended to extract the errors used for mass calibration. The module will include any consecutive matched error from the most populated error range found within the established error range tolerance (i.e., 0.1PPM). This process continues until the range cannot be extended anymore according to the parameter set.

Example:

#### MASS CALIBRATION METHOD

 To estimate mass measurement bias more accurately, we can model the trend exhibited by the error size vs m/z value relation obtained by matching the mass peaks. With the estimation model we can shift/calibrate the mass peaks at different particular m/z values more accurately.

The module supports two main modes for mass calibration:

- Arithmetic mean:

This method uses the arithmetic mean of the extracted errors of the overall bias estimation.

Calibration of peaks will be performed globally based on a single overall bias value. This method is especially recommended for datasets with low number of extracted errors (i.e., blank samples).

- Regression mode:

This mode models a trend from the direct relationship of error size (in PPM) vs. measured m/z of detected peaks. Mass lists will be calibrated according to the estimated model. The mass calibration module supports two different methods of regression: **OLS** and **KNN**.

- **OLS (ordinary least squares)** regression minimizes the mean squared error between the predicted trend and the datapoints in the dataset. Available features include power features (polynomial trend), logarithmic feature and exponential feature. **By default**, linear trend is fitted. This mode is suitable for datasets with enough data and exhibit a clear and strong trend (Error vs. m/z).

- **KNN (K-Nearest Neighbors)** regression finds the average value of the K nearest neighbors. In this module, the number of neighbors is defined by a percentage set by the user of all the errors present in the dataset. The K closest neighbors are thus found by the absolute difference of the m/z values within such percentage. Then the arithmetic mean of the neighbors' errors is calculated for each individual error and will serve as an error estimate for a specific m/z. This method is suitable for datasets with enough data and a trend between mass error vs. m/z is not clear. Therefore, KNN regression allows the trend to match the dataset closely without introducing additional assumptions on how the variables are related.

 Overfitting problems at large m/z values (>800) can occur when modeling the errors with regression as those regions commonly have less matches against the reference libraries of ions. If regression is modelled mainly by matches with small m/z ions (<400), it is recommended to use the arithmetic mean for mass calibration. This also applies to blank samples where the number of matched ions is commonly low and regression can produce overfitting problems.

#### Parameters

##### Raw data files

The raw datafiles to calibrate. Each selected datafile is calibrated independently in a separate task.

**Mass list name**

Name of the mass lists to be calibrated. The mass lists must be previously generated for each scan by the Mass detector module.

**Intensity threshold**

Determines the intensity of the peaks used for matching against the library of ions and subsequent mass calibration. Only mass peaks with intensity above this threshold will be considered for calibration. This is useful to avoid certain noise peaks that could have been picked in the Mass Detection module. To consider all peaks, the Intensity Threshold needs to be set with a value equal or lower than the previously used in the Mass Detection module.

**Duplicate Error Filter**

Removes duplicate ions with the same m/z value independently of their retention time. If enabled, for a specific detected exact mass present in different scans (not necessarily consecutive), only a single ion with that exact mass value will be considered for calibration. This filter performs for the full list of masses and does not consider RT difference between ions.

**Reference Library of ions**

Selects the library used for ion matching and determination of mass errors. SCL and UCL libraries are available.

- **SCL-only parameters**

*Standard Calibrant Library file*

File with a list of ion formulas and retention times (xls, xlsx and csv files are supported). This list should contain ions that are expected to be detected in the samples. Files need to contain a first column with the retention time in minutes and a second column with the ion formula strings. Additional columns are optional. Sample standards list file:

*Retention time tolerance (only for SCL)*

Maximum difference in retention time between an actual measured ion and a calibrant to consider a match.

- **UCL-only parameters**

**Ionization mode**

Ionization mode for which an appropriate universal calibrants list is used.

**m/z tolerance**

Maximum allowed difference in m/z between an actual measured ion and a calibrant to consider a match.

**Overall Mass Bias Estimation**

Method used to determine the proper range of errors to perform calibration.

If both parameters (*Primary High-Density Range of Errors size* and *Error range tolerance*) are set to zero, all errors obtained after matching against the reference library of ions are used for calibration of peaks.

- **High-Density Range of Errors**

*Primary High-Density Range of Errors size*

Determine the range (in PPM) containing most mass errors after matching the detected ions with the ion calibration library. Use zero to skip this step, in such case the distribution is split into subranges containing all the errors within the error tolerance and the largest subrange is used.

*Error Range Tolerance*

Maximum distance (in PPM error) between the maximum and minimum thresholds of the Primary High-Density Range of Errors and the consecutive error to allow the extension of the error range. This process continues until the algorithm does not find any consecutive error within the Error Range Tolerance value. Use zero to skip this step and no extension will be computed.

- **Percentile Range of Errors**

*Percentile range*

Top and bottom percentiles to define the range of errors (Interquartile range) used for mass calibration.

## Mass Calibration method

Method used for mass calibration.

- **Arithmetic mean**

- **KNN regression**

- *Nearest neighbors' percentage*

Percentage of nearest neighbors used for error prediction. - **OLS regression**

- **Polynomial degree**

The degree of polynomial trend used, the summand powers of the polynomial will be the OLS regression features. Use 0 for constant component, 1 for linear, 2 for quadratic and so on.

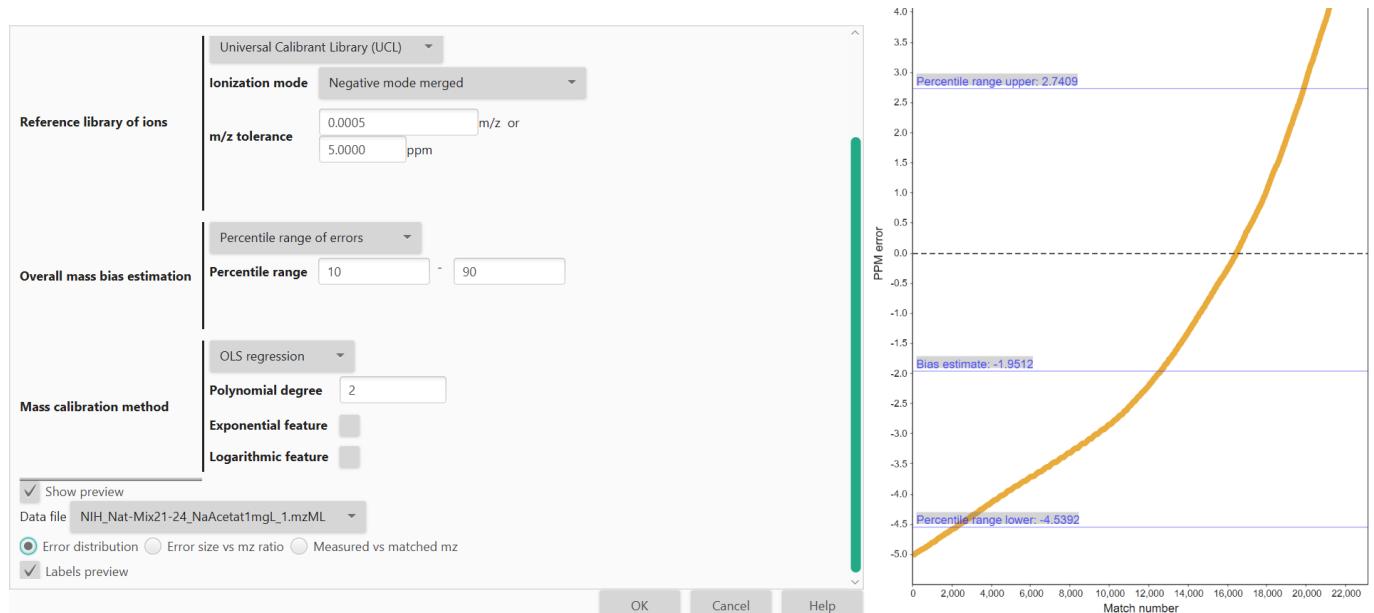
- *Exponential feature*

When selected, an exponential feature  $\exp(x/10)$  is included.

- *Logarithmic feature*

When selected, logarithmic feature  $\ln(x)$  is included.

## Examples



This module was initially created during a GSoC 2020 project with MZmine by Łukasz Fiszer and MZmine team.

Last update: August 1, 2022 17:24:00

## 5.3 Mobility scan merging

---

If a .mzML file is imported, the merged frame spectrum must be created via the **File merging → Mobility scan merging** module. This is required to gain access to MZmine's regular LC-MS functionality. This step uses the centroided and thresholded data produced by the [mass detection](#) step.

**This step is not required when importing native Bruker .tdf or .tsf data from .d folders.** When importing native Bruker data, a merged spectrum for the frame is created automatically by the vendor library.

---

Last update: April 5, 2022 11:25:19

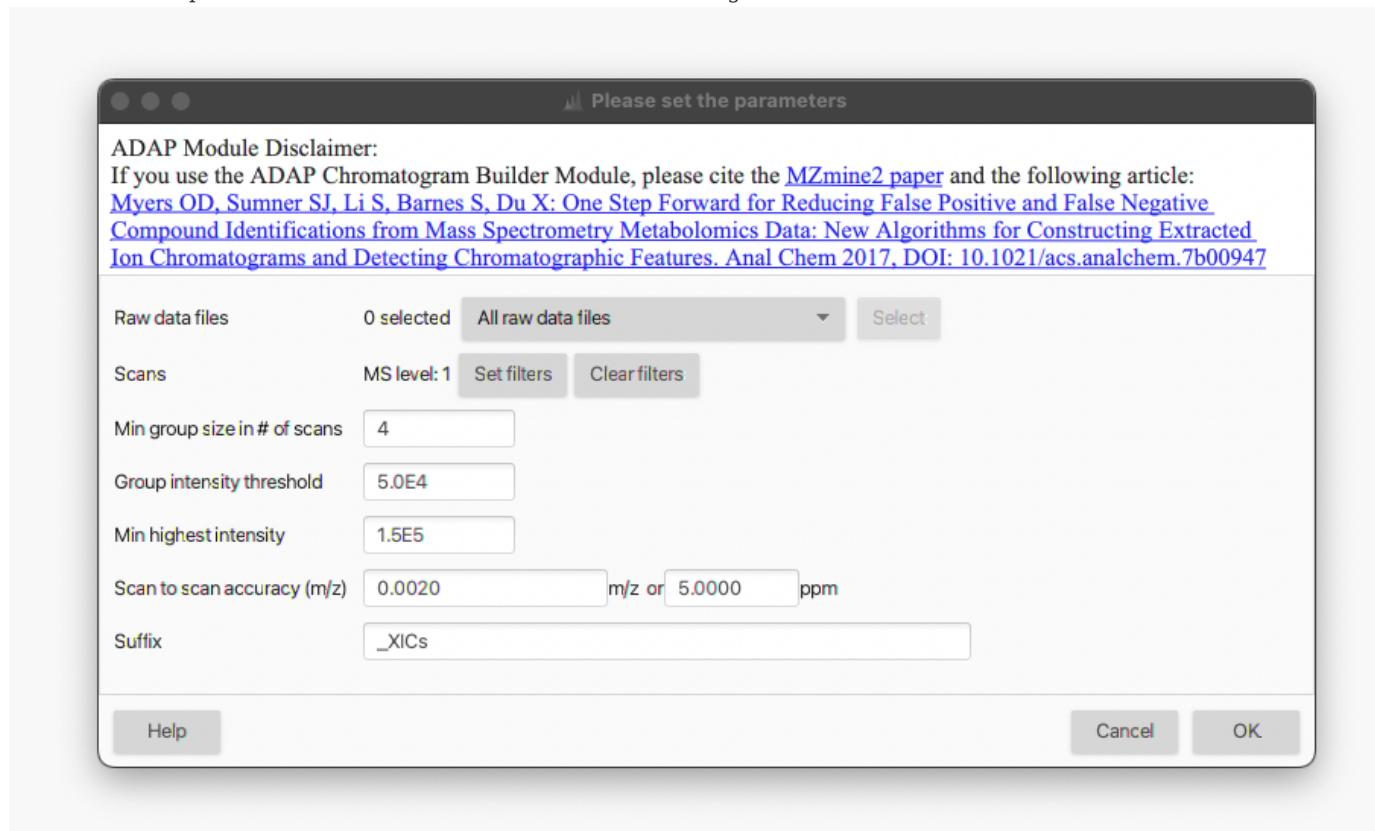
## 5.4 ADAP chromatogram builder

The *ADAP chromatogram builder* module is one of the LC-MS feature detection algorithms provided by MZmine 3. The module essentially builds an EIC for each  $m/z$  value that was detected over a minimum number of consecutive scans in the LC-MS run. Each data file is processed individually. The [mass list](#) associated to each MS1 scan in a data file (see [Mass detection](#) module) are taken as input and a [feature list](#) is returned as output. Since a mass list must be available, the *Mass detection* module must be run first.

The *ADAP chromatogram builder* algorithm operates as follows. Only MS1 scans are processed. All the data points are extracted from all the MS1 scans in a data file and sorted in order of decreasing intensity. The processing starts from the most intense data point and, since no EICs have yet been created, a new EIC is initialized and associated to the corresponding  $m/z$  value. The processing proceeds with the second-highest data point and the corresponding  $m/z$  is checked to determine if it "belongs" to the existing EIC based on the user-defined tolerance (*i.e.* "Scan to scan accuracy ( $m/z$ )" parameter). If so, the data point is added to the EIC and the EIC-associated  $m/z$  is updated. Otherwise, a new EIC is initialized. The process is iterated until all the data points have been processed and a set of EICs has been created. Finally, the EICs are checked according to the user-defined parameters (*i.e.* minimum number of data points and intensity). The EICs matching the requirements are retained in the *feature list*, whereas the rest are discarded. The so-built EICs can then be resolved into individual features by one of the deconvolution algorithms provided by MZmine 3 (*e.g.* [Local minimum resolver](#) module).

### 5.4.1 Parameters settings

:material-menu-open: Feature detection → LC-MS → ADAP chromatogram builder



#### RAW DATA FILES

Select the input raw data files for chromatogram building. Mass lists associated with the data files will be automatically selected. See option descriptions in [Mass detection](#) module.

**SCANS**

Select (or filter out) the MS scans to be processed. Although setting the *MS level = 1* is usually sufficient for this module, several filters are available (see option descriptions in [Mass detection](#) module). For example, specific RT ranges (*e.g.* dead volume, equilibration time, calibration segments, *etc.*) can be excluded from the processing by setting the corresponding filter.

**MIN GROUP SIZE IN NUMBER OF SCANS**

Minimum number of consecutive MS1 scans where a *m/z* must be detected with a non-zero intensity in order for the corresponding EICs to be considered valid and retained in the feature list.

**💡 Tip.** This parameter largely depends on the chromatographic system setup (*e.g.* HPLC vs UHPLC) and the acquisition rate (*a.k.a.* MS scan speed) of the mass spectrometer. The best way to optimize this setting is by manually inspecting the raw data and determining the typical minimum number of data points of the LC peaks. Usually, no less than 4-5 should be used.

**GROUP INTENSITY THRESHOLD**

Minimum signal intensity that the group scans (see previous parameter) must exceed in order for the corresponding EICs to be considered valid and retained in the feature list.

**💡 Tip.** A good starting point for this parameter is 3 times the noise level used in the [Mass detection](#), if the instrumental noise is used as cutoff. See also [How do I determine the noise level in my data?](#) for more details.

**MIN HIGHEST INTENSITY**

Minimum intensity that the highest point in the EIC must exceed in order for the corresponding trace to be considered valid and retained in the feature list. This parameter mainly depends on the mass spectrometer characteristics (*e.g.* Orbitrap instruments normally provides higher signal intensities than TOF devices) as well as the overall goal of the processing. Overly low intensity thresholds normally leads to a larger number of background signals being retained as features, extending the overall processing time. On the other hand, overly high thresholds may lead to low-intensity features being erroneously discarded. **💡 Tip.** A good starting point for this parameter is 7-10 times the noise level used in the [Mass detection](#), **if** the instrumental noise is used as cutoff. See also [How do I determine the noise level in my data?](#) for more details.

**SCAN TO SCAN ACCURACY (M/Z)**

Maximum allowed difference between an EIC-associated *m/z* and a new data point to be added to the existing EIC trace. It is essentially the maximum allowed mass accuracy deviation between consecutive data points in the EICs. The tolerance can be specified as absolute tolerance (in *m/z*), relative tolerance (in ppm), or both. When both are specified, the tolerance range is calculated using the maximum between the absolute and relative tolerances.

**💡 Tip.** This is an [inter-scan \*m/z\* tolerance](#) and it depends on the mass accuracy, resolution and stability of the instrument. The best way to optimize this parameter is by manually inspecting the raw data and determining the typical fluctuation of the accurate mass measurement over consecutive scans. A good starting point is 0.002-0.005 *m/z* and 5-10 ppm for Orbitrap instruments, while 0.005 *m/z* and 10-15 ppm can be used for TOF devices.

**SUFFIX**

String added to the filename as suffix when creating the corresponding feature list.

Last update: July 22, 2022 07:54:38

## 5.5 IMS Expander

The IMS expander will search for data points in mobility scans for existing features. This requires prior chromatogram building (see [ADAP Chromatogram builder](#) and resolving in retention time dimension (see [Resolving](#)).

### Parameter settings

#### M/Z TOLERANCE

If selected, a tolerance will be applied to the feature's detected m/z while searching for data points in mobility dimension. Otherwise, the accepted m/z range is determined by the feature's m/z distribution in accumulated frame spectra.

**Recommended setting: selected, 0.003 m/z and 15 ppm**

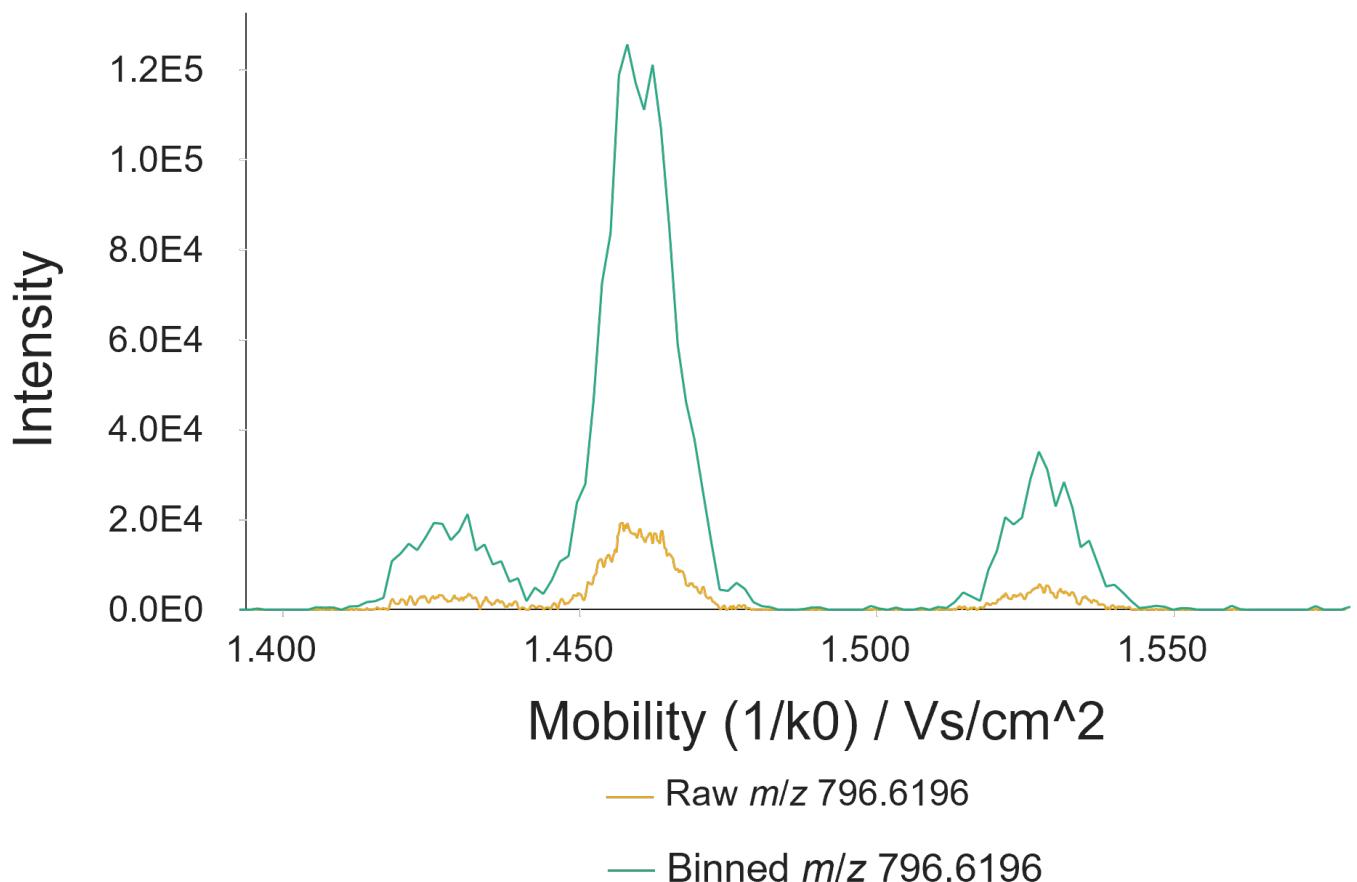
#### RAW DATA INSTEAD OF THRESHOLDED

Enables searching in mobility scan raw data instead of the thresholded (=mass detected) data. Only possible for centroid raw data files.

#### OVERRIDE DEFAULT MOBILITY BIN WIDTH (SCANS)

If selected, the default number of binned mobility scans can be overridden. Useful for data with high mobility resolution.

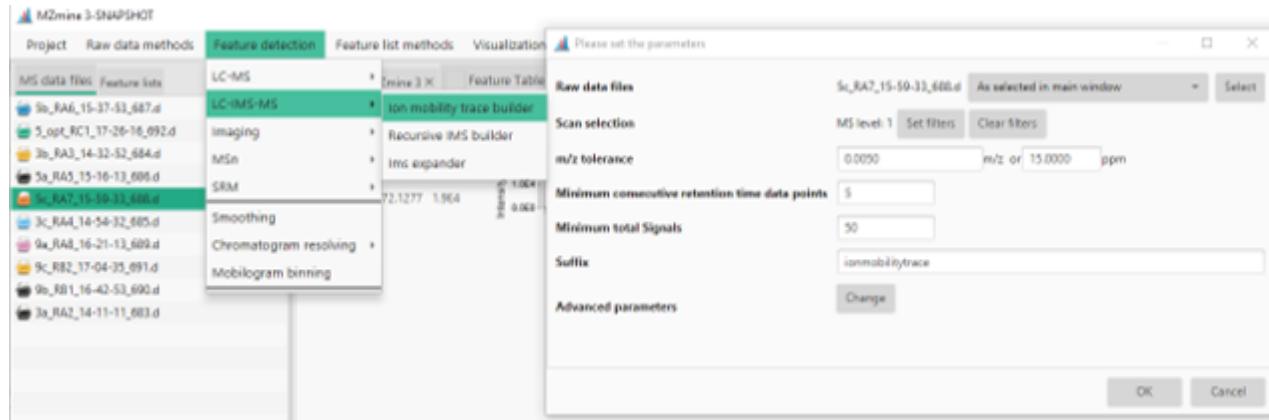
### Binned mobilogram example



## 5.6 Ion mobility trace builder

The **Ion mobility trace builder** will build ion mobility traces from the raw data. Alternatively, the **Recursive IMS builder** can be used, which requires less ram but takes longer.

The **Ion mobility trace builder** is found under **Feature detection → LC-IMS-MS → Ion mobility trace builder**.



### Scan selection

The scan selection parameter specifies the scans that shall be processed for feature detection. Usually, setting the ms level to 1 is sufficient. If a calibration segment is present, it can be cut out via the retention time filter in the scan selection.

### m/z tolerance

The **m/z tolerance** specifies the scan-to-scan tolerance for ion mobility traces. This tolerance window may need to be set higher than for classic LC-MS feature detection (e.g. to 0.005 m/z and 15-20 ppm instead of 10 ppm) due to lower intensities therefore less accuracy in individual mobility scans compared to LC-MS scans. Note that the overall accuracy is achieved via LC-IMS-MS is the same due to the higher number of scans.

### Minimum consecutive retention time data points

This parameter specifies the number of consecutive detections of the same m/z value in a chromatographic peak (rt dimension only). This means that a single m/z has to be detected in, e.g. 5 frames with an intensity higher than zero. This parameter helps to filter noise. Consecutive detections in the mobility dimension do not affect this parameter. Usually no less than 5 should be set here if the MS1 acquisition rate is sufficient.

### Minimum total signals

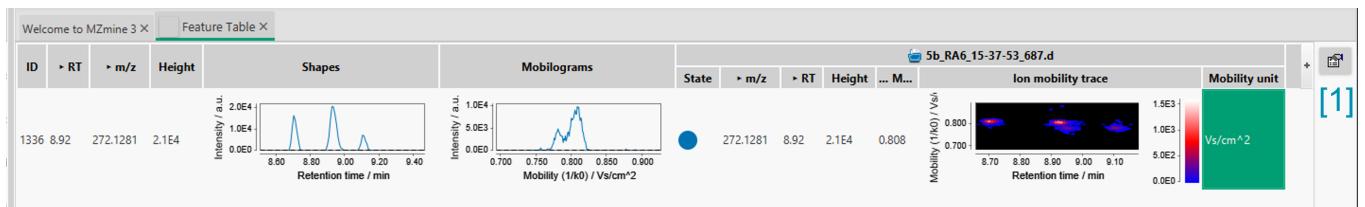
Specifies the total number of peaks in the mobility dimension in all mobility scans. Every "dot" in an ion mobility trace represents a single datapoint. ( see [ion mobility traces](#) )

### Advanced parameters

For most applications, these parameters do not need to be set/changed. For high mobility resolved data the mobilograms might become noisy due to less ions reaching the detector at the same time. By default, the number of binned scans is set to cover about 0.0008 Vs/cm<sup>2</sup> per bin. The effect of binning can be seen [here](#). If you are unsure about the nature of your data, you can perform trace building with the standard parameters and apply/preview the binning afterwards via the **Feature detection → Mobilogram binning** module.

### Processing result

After performing ion mobility trace detection, a feature list is created in the feature list tab (see [feature lists tab](#)). In the feature table, multiple columns are created. The displayed columns can be set via the button on the right of the feature table ([1]).



The **shapes** displays an EIC of the ion mobility trace (intensities summed in rt dimension). The **mobilograms** column shows a mobilogram for the ion mobility trace (intensities summed in mobility dimension). The shapes and projections can be smoothed and resolved. However, the ion mobility trace is always represented by the raw data and remains unaltered. After resolving, the shapes and mobilograms have to be recalculated from the raw data, which is why the smoothing is lost after resolving.

Last update: April 5, 2022 13:31:49

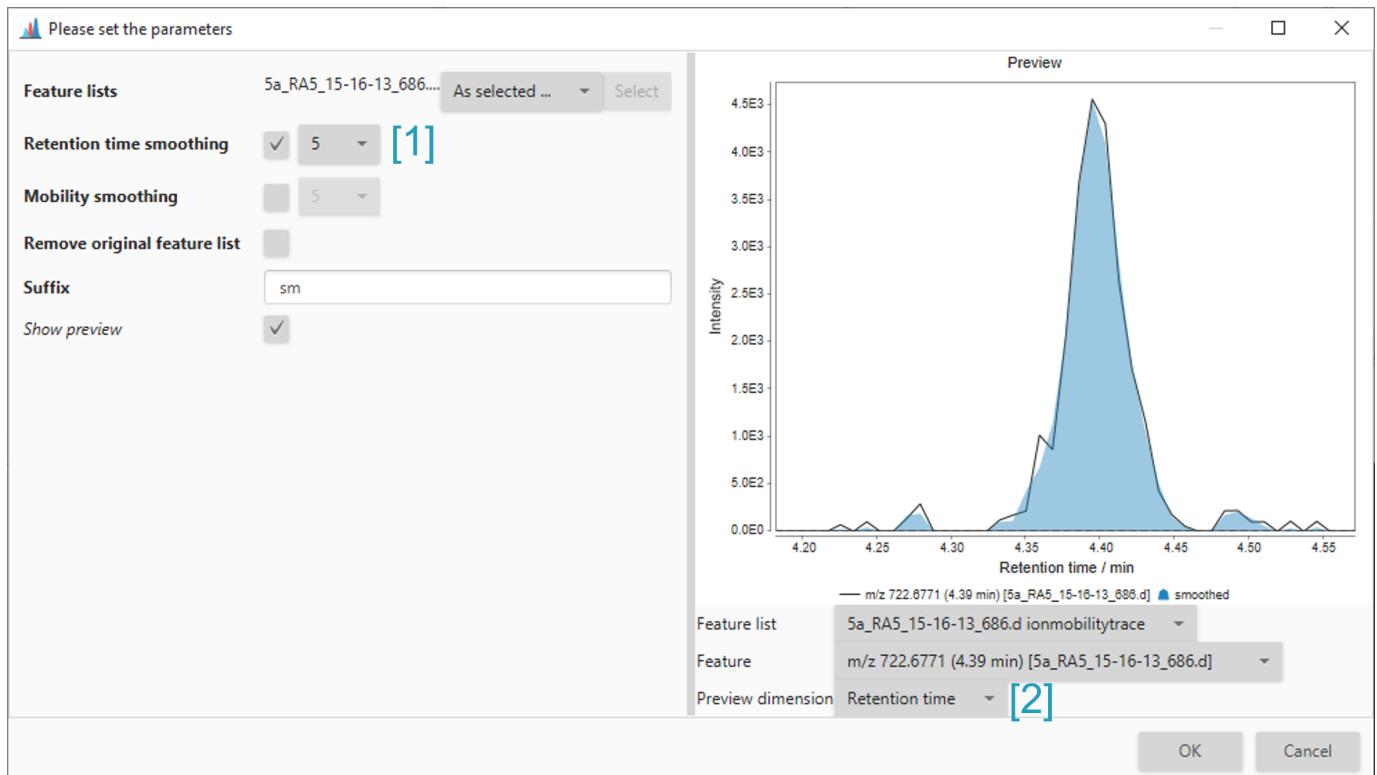
## 5.7 Smoothing

### 5.7.1 Retention time dimension

Smoothing chromatograms is optional. The necessity of smoothing in RT dimension is determined by the noisiness of chromatographic peaks. These can be influenced by the overall spray stability, instrument accumulation times, transfer efficiency and many more.

The number of data points to be smoothed in rt dimension can be set at [1]. Note that the correct preview dimension is selected at [2].

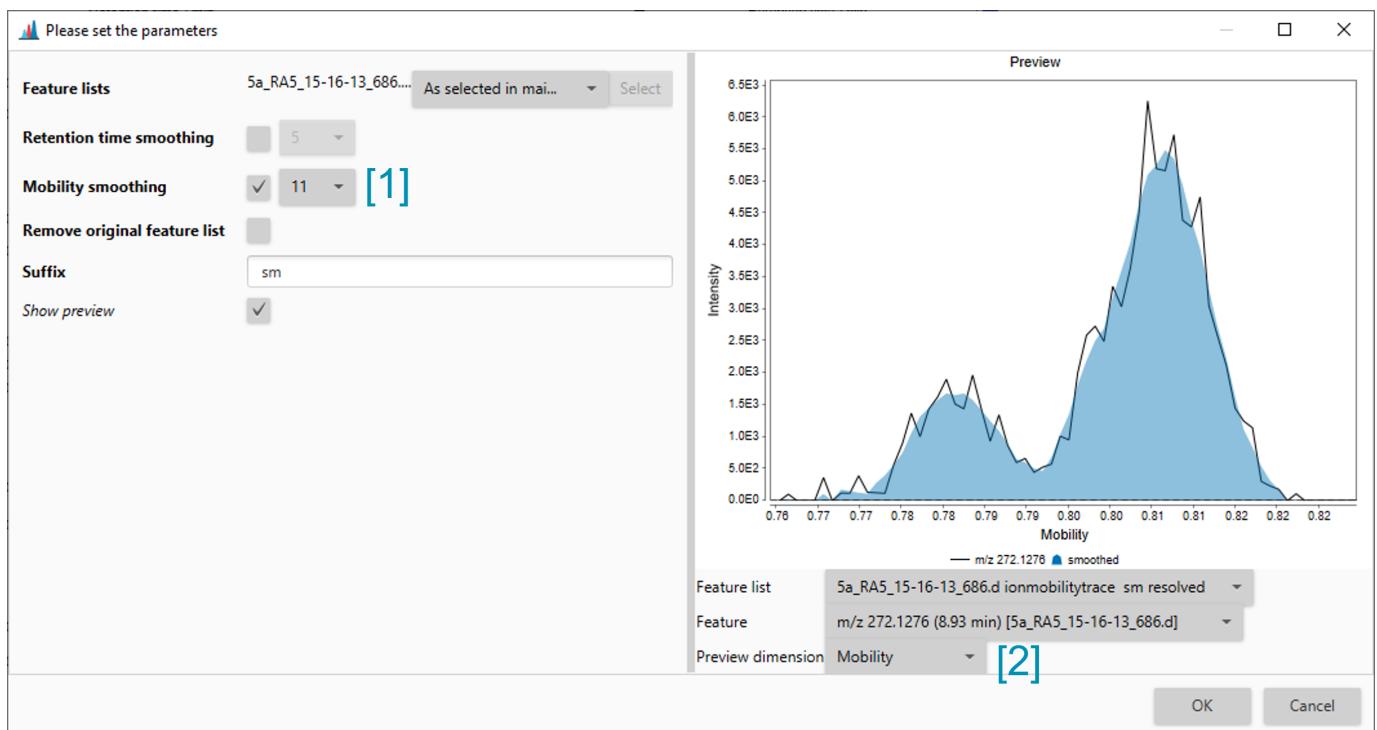
For large batch modes, the **Remove original feature list** parameter should be selected. While parameters are being optimised, this is not recommended, because removing a feature list cannot be undone.



### 5.7.2 Mobility dimension

After resolving a feature in RT dimension, the mobilograms will be recalculated from the raw data (the resolved ion mobility trace). Therefore, a smoothing step is necessary if the data requires it. The smoothing dialog is opened via **Feature detection → Smoothing**

Select to smooth the mobility dimension [1] and select it as preview dimension [2]. The filter with depends on the number of spectra acquired in the observed mobility range. Usually, a value between 5 and 15 should be appropriate.



Last update: March 10, 2022 15:42:38

## 5.8 Local Minimum Resolver

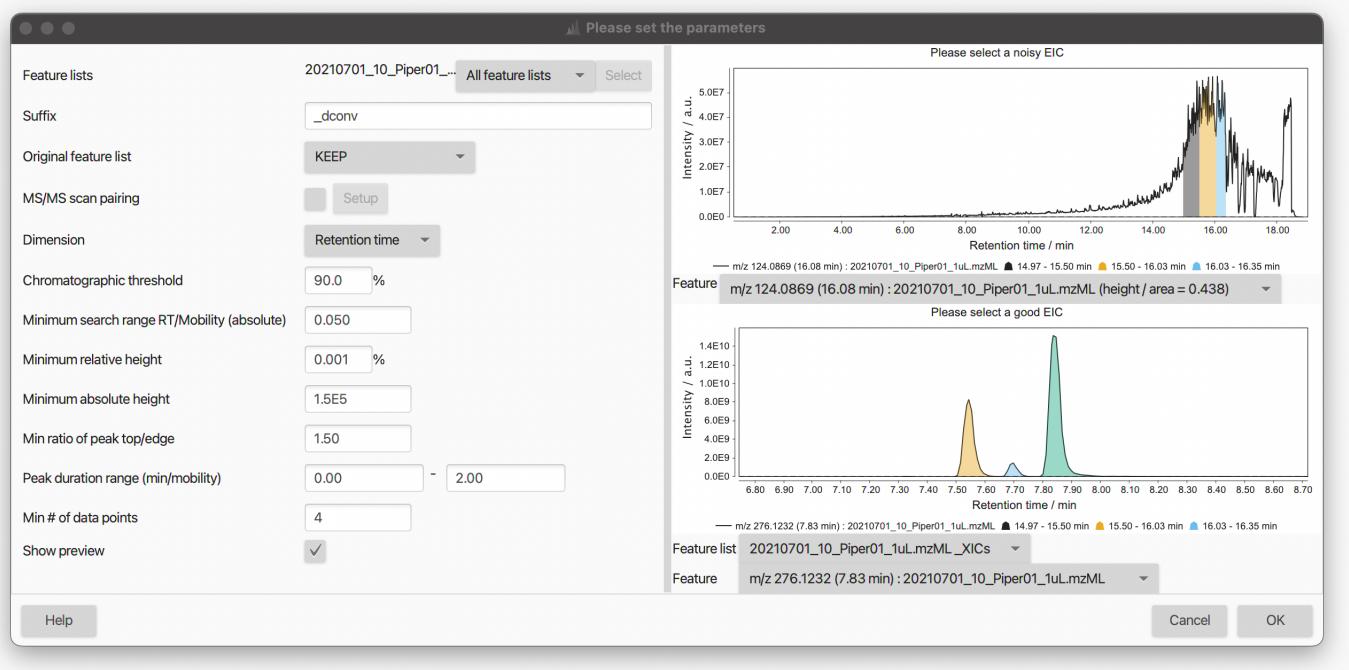
During the EICs building, overlapping and partially co-eluting peaks are retained as single features in the feature list (see, for example, [ADAP chromatogram builder](#)). The *local minimum resolver* module aims at splitting such "shoulder" LC peaks into individual features (*i.e.* [chromatographic resolving](#)) based on local minima. In fact, a local minimum in the EIC trace might correspond to the valley between two adjacent, partially-resolved peaks.

The algorithm examines all the data points in the EIC trace starting from the earliest RT. A scan window is set (see [Minimum search range RT/Mobility](#) parameter) and centered around the examined data point. A data point is considered a local minimum if it is the lowest intense point within the scan window. When a local minimum is found, a set of user-defined intensity and peak duration requirements is checked. If they are fulfilled, the original overlapping peaks are split into new, distinct features. The LMR is particularly suitable for LC-MS data with little noise and nice peak shapes.

With the implementation of ion mobility (IM) support in MZmine 3, this module was expanded and can now be applied over both the RT and IM dimensions (see [Resolving the ion mobility dimension](#)).

### 5.8.1 Parameters settings

:material-menu-open: Feature detection → Chromatogram resolving → Local minimum resolver



#### SUFFIX

String added as suffix to when creating the new feature list(s).

#### ORIGINAL FEATURE LIST

Keep or remove the input feature list(s). The *PROCESS IN PLACE* option directly filter the input feature list and performs better in terms of memory usage; therefore, it is recommended over *REMOVE*, when available.

#### MS/MS SCAN PAIRING

Pair MS/MS fragmentation spectra collected in [DDA](#) mode to the resolved features. This is optional at this stage as the same can be done later in the pipeline using a separate [module](#). See [MS2 scan pairing](#) documentation for more details.

**DIMENSION**

Dimension to be resolved. Select *Retention time* or *Mobility* to run the module over the RT or IM dimension, respectively.

**CHROMATOGRAPHIC THRESHOLD**

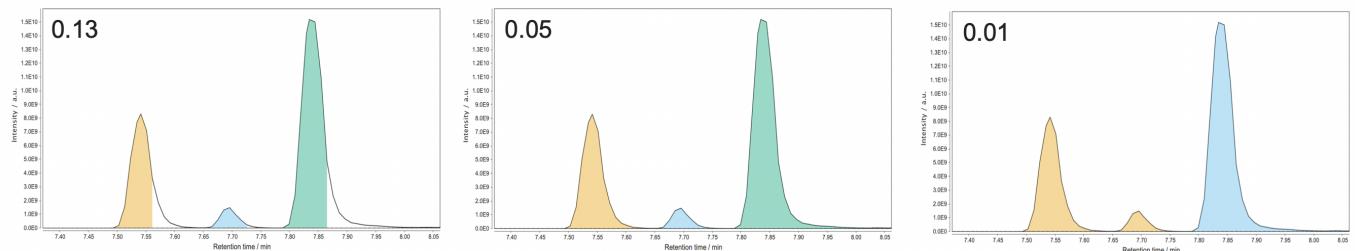
Percentage of data points in the EIC removed before local minima search. This represents an important filter for noisy chromatogram and significantly reduces the precessing time. The algorithm finds the intensity value (threshold) that leaves the specified percentage of data points in the EIC trace below XXX. All such data points are removed. For example, a *Chromatographic threshold* = 50% will discard the lowest-intense 50% data points in the EIC trace.

**Tip.** It must be noted that the algorithm examines the EICs throughout the entire RT range (*i.e.* also the zero data points are considered). Therefore, we recommend to set this value rather high (*e.g.* 90-95%) and lower it only if needed.

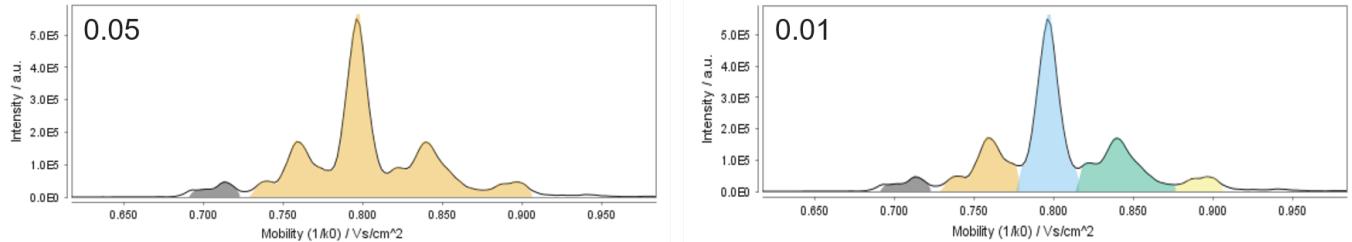
**Tip.** When [resolving the ion mobility dimension](#), we recommend to lower this settings to no more than 80% since [mobilograms](#) contains less data points than regular LC traces.

**MINIMUM SEARCH RANGE RT/MOBILITY (ABSOLUTE)**

Size of the RT, or mobility, window examined for local minimum search. An overly narrow search range can cause peak edges to be cut off, whereas a too wide search ranges might lead to an incomplete resolution of narrowly eluting peaks (see example below).



**Tip.** A shorter *Minimum search range* is generally needed when [resolving the ion mobility dimension](#).

**MINIMUM RELATIVE HEIGHT**

Minimum relative intensity (respect to the highest data point in the EIC) a peak need to reach to be retained as a feature. This parameter can be used in combination with the [Minimum absolute height](#) setting as to filter the resolved features to be retained. Many users prefer to rely only on the [Minimum absolute height](#) as it is more straightforward to set. To do so, set the *Minimum relative height* = 0 and the parameter will be ignored.

**Tip.** Modern mass spectrometers provides linear dynamic ranges up to 5 orders of magnitude. If we take an Orbitrap device with a detector saturation around 1.0E10 intensity, a *Minimum relative height* = 0.001 would correspond to a 1.0E5 minimum intensity.

**MINIMUM ABSOLUTE HEIGHT**

Minimum absolute intensity a peak needs to reach to be retained as a feature. This parameter is very similar to the [Min highest intensity](#) settings in the ADAP chromatogram builder module and the same concepts apply.

**Tip.** When resolving the RT dimension, the same value used as [Min highest intensity](#) in the EICs building can normally be used here.

**Tip.** While [frame scans](#) are examined over the RT dimension, [mobility scans](#) are examined over the IM dimension. Therefore, this parameter might need to be adjusted accordingly when [resolving the ion mobility dimension](#).

**MIN RATIO OF PEAK TOP/EDGE**

Minimum ratio between the intensity of the highest (apex) and side (left and right 'edges') points of a peak, to retain it as a feature. The peak edges have to be X times less intense than the peak apex for the feature to be retained. The purpose of this parameter is to reduce the detection of false local minima when the exaimend trace (EIC or *mobilogram*) is not smooth. In general, this mainly affects low intensity and not-baseline-resolved signals

**Tip.** This parameter can best be optimized using the *Show preview* option. We recommend values between 1.7 (not baseline separated) and 2 to start the optimisation.

**PEAK DURANTION RANGE (MIN/MOBILITY)**

Range of acceptable peak length expressed in minutes (RT dimension) or absolute units (mobility dimension). This parameter can be used to filter out noisy features based on their overly short (or long) duration.

**MIN # OF DATA POINTS**

Minimum number of data points a resolved peak needs to have to be considered valid and retained as a feature. This parameter can be used along with the [Peak duration range](#) setting as peak duration constraint to filter out noisy features.

**Tip.** This parameter is very similar to the [Min group size in # of scans](#) settings in the ADAP chromatogram builder module and the same value can normally be used here (usually, no less than 4-5).

**Tip.** A feature in the IM dimension is normally made up of more data points than regular LC peaks. Therefore, a higher *Min # of data points* can be set when [resolving the ion mobility dimension](#) to filter out noisy features.

**SHOW PREVIEW**

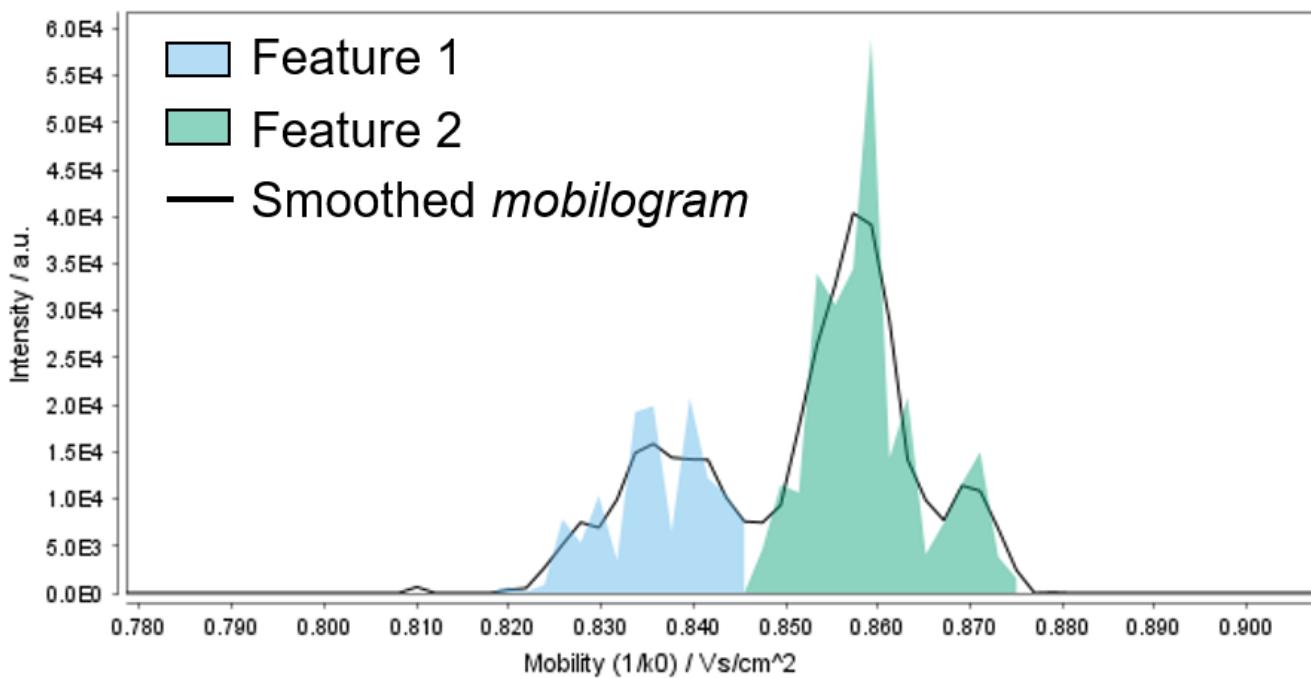
By checking this box, an interactive visualization panel will open to help the user to adjust the algorithm parameters. Two EIC traces can be displayed simultaneously in two sub-panels to assess the impact of chosen settings on both "good" and "noisy" EIC traces. The feature list and EIC traces to display can be selected from the corresponding drop-down menus. A noisy EIC can generally be found by sorting the feature table by decreasing area, or by looking at the height/area ratio provided for each feature in the top sub-panel (noisy EIC tend to have low height/area ratios). We recommend optimising the parameters on good EICs and checking the results of these parameters with a noisy EIC.

---

## 5.8.2 Resolving the ion mobility dimension

This same module can be used to resolve features co-eluting in the RT dimension, based on their ion mobility. The same concepts apply as in the resolution of the RT dimension. However, *mobilograms* are examined instead of EIC traces and the same settings used for the RT dimension might not be optimal when resolving IM data. In particular, the following aspects should be born in mind:

1. While [frame scans](#) are examined over the RT dimension, [mobility scans](#) are considered over the IM dimension. As explained [here](#), *frame scans* are essentially obtained by merging the *mobility scans* acquired over an IM accumulation. Therefore, it might be necessary to adjust parameters like [Minimum absolute height](#) or [Min ratio of peak top/edge](#) to account for the lower signal intensity of *mobility scans*.
2. [Mobilograms](#) are recalculated from raw data, even though a [smoothing](#) step was previously applied. Non-smoothed *mobilograms* tend to be more jagged than regular EIC traces (see Figure). Threfore, some parameters (e.g. [Min search range](#) and (e.g. [Min ratio of peak top/edge](#)) should be adjusted accordingly.



1. Mobilograms contain less scans (e.g. ≈400-1000 per frame, depending on instrument type and acquisition settings) compared to regular EICs (e.g. ≈4500 scans for 15 minutes LC run and scan rate of 0.2 seconds). Therefore, a lower chromatographic threshold (e.g. 80%) is recommended to avoid relevant data points in the mobilogram being discarded.

On the other hand, a single feature in the IM dimension is normally made up of more data points than regular LC peaks, due to the different timescale the IM separation is performed on (see [here](#) for more details). Therefore, a higher Min # of data points can be set when [resolving the ion mobility dimension](#) to filter out noisy features.

2. Different vendors use different units of mobility. For instance, [TIMS](#) express ions' mobility as Vs/cm<sup>2</sup>, whereas [time dispersive IM devices](#) (DTIMS and TWIMS) use the ions' drift time (expressed in milliseconds). TIMS values are numerically smaller than DTIMS or TWIMS; therefore, the [minimum search range](#) parameter should be adjusted accordingly.

---

Last update: June 23, 2022 22:05:12

## 5.9 CCS Calibration and calculation

Accurate determination of CCS values requires a valid CCS calibration and molecule charge states to be detected.

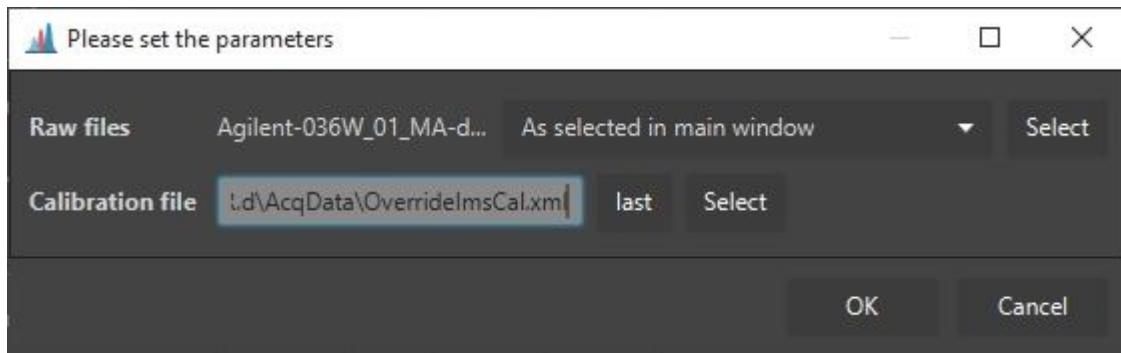
- **timSTOF** raw data can be recalibrated using data analysis and imported in MZmine. The recalibrated data will be used by default. (see [Calculating CCS values](#))
- **mzML** raw data requires the determination of a calibration function from the raw data (e.g. as detected features) or as import from an external file. (see [Creating or importing a CCS calibration](#))

### 5.9.1 Creating or importing a CCS calibration

#### Importing a CCS calibration

Agilent calibration data can be imported from the "OverrideImsCal.xml" file in the Agilent raw data folder. Waters calibration data can be imported from the "mob\_cal.csv" file in the Waters raw data folder. The "\_extern.inf" file is also required, but will be read automatically when the "mob\_cal.csv" is selected.

The calibration import is accessed via **Feature list methods -> Processing -> External CCS Calibration**. Then select the calibration "OverrideImsCal.xml"/"mob\_cal.csv" from the raw data folder, and select the raw data files the calibration should be applied to.

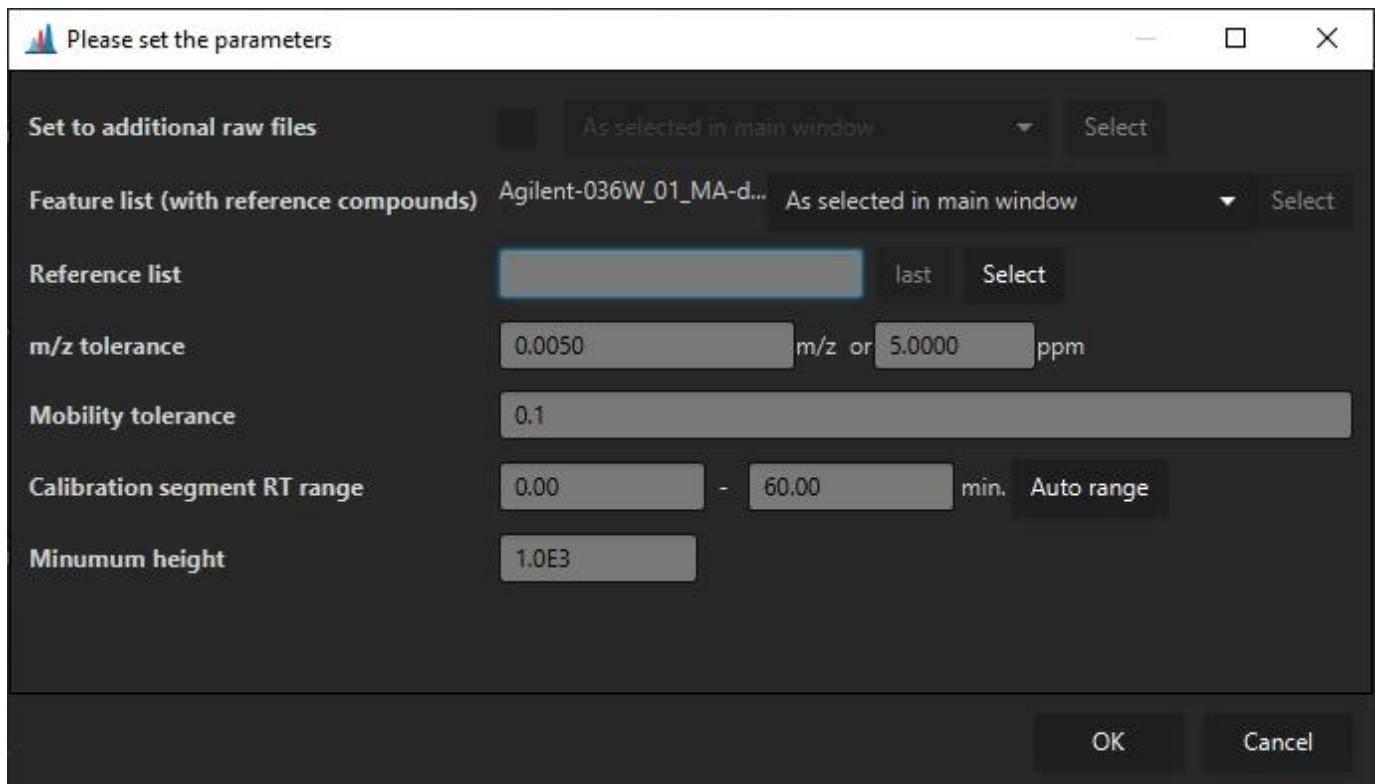


#### Reference CSS calibration

If a mobility calibrant is infused during an HPLC run of every sample, a CCS calibration can be calculated on a per-raw file basis. (Common procedure on Bruker devices) Otherwise, a single run can be used to calibrate multiple files.

Please note that this is currently only supported for TIMS and DTIMS data.

The calibration module can be accessed via **Feature list methods -> Processing -> Internal reference calibration**.



**Set to additional raw files** If a calibration calculated from a single feature list shall be applied to multiple other raw files, the raw files can be selected here. This requires only a single raw file to be selected.

**Feature list (with reference compounds)** Specifies (a) feature list(s) that contains the reference compounds. If multiple feature lists are selected, every feature list will be searched for reference compounds, and the calibration will be used for the raw data files in the particular feature list. This means that no raw data file may be selected. (Cannot set multiple calibrations to a single raw file.)

If a single feature list is selected, the calibration may be applied to additional raw data files via the **Set to additional raw files** parameter.

**Reference list** Specifies a ".csv" reference list of for CCS calibrant ions. Must contain the columns "mz", "mobility", "ccs", "charge". Columns must be separated by ";". The ion mode may be specified via the charge of the ion, e.g., as 1 or -1. Only the correct polarity will be used to calculate the calibration.

**m/z tolerance** The m/z tolerance for the reference compounds.

**Mobility tolerance** the mobility tolerance to detect the reference compounds.

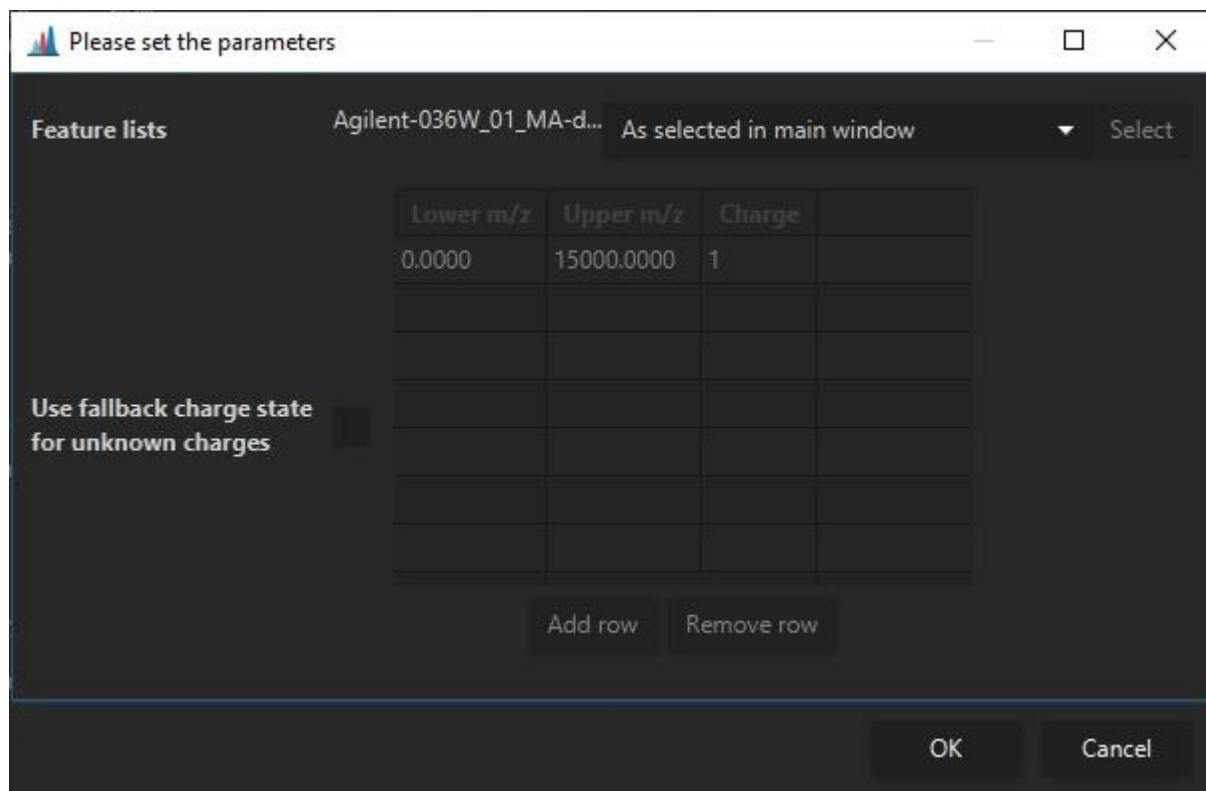
**Calibration segment RT Range** Specifies the rt range that shall be searched for calibrant ions. Usually either the beginning or end of a HPLC run.

**Minimum height** A minimum intensity for reference compounds to be used as calibrant signals for determination of the calibration.

## 5.9.2 Calculating CCS values

After a calibration as been set (Agilent/Waters/Bruker mzML) (Bruker tdf works out-of-the-box) CCS values can be calculated via **Feature list methods -> Processing -> Calculate CCS values**.

Here, a default charge state may be set, in case it could not be determined. Otherwise, the charge state determined via the isotope pattern will be used.



Last update: March 10, 2022 15:42:38

## 5.10 Gap-filling

---

Some chromatographic peaks may not be detected in every sample for several reason: - Reason 1 - Reason 2 - Reason 3

- This causes undesirable gaps (missing values) in the aligned feature table.
  - To tackle this issue, a value for the peak needs to be imputed
  - A simple gap-filling approach is to integrate the area where the peak is expected but not detected
  - These areas usually correspond to spectral noise
  - By doing so, no bias is introduced
  - The gap-filled feature table can now be used in downstream data analysis
- 

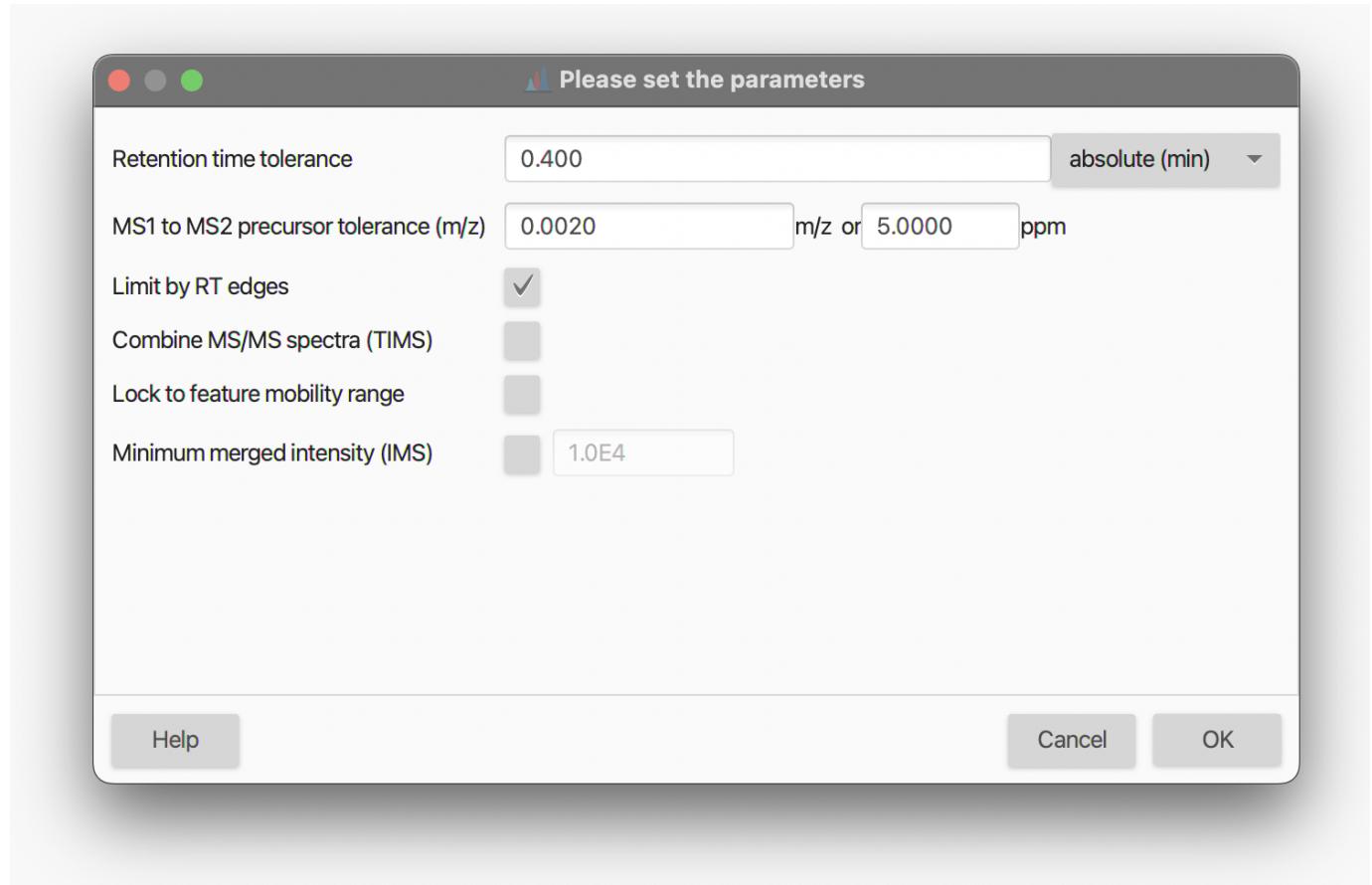
Last update: March 10, 2022 15:42:38

## 5.11 MS2 Scan Pairing

Description

### 5.11.1 Parameters settings

:material-menu-open: Feature list methods → Processing → Assign MS2 to feature



#### RETENTION TIME TOLERANCE

To-do

#### MS1 TO MS2 PRECURSOR TOLERANCE (M/Z)

To-do

#### LIMIT BY RT EDGES

To-do

#### COMBINE MS/MS SPECTRA (TIMS)

To-do

#### LOCK TO FEATURE MOBILITY RANGE

To-do

#### MINIMUM MERGED INTENSITY

To-do

Last update: April 5, 2022 13:22:51

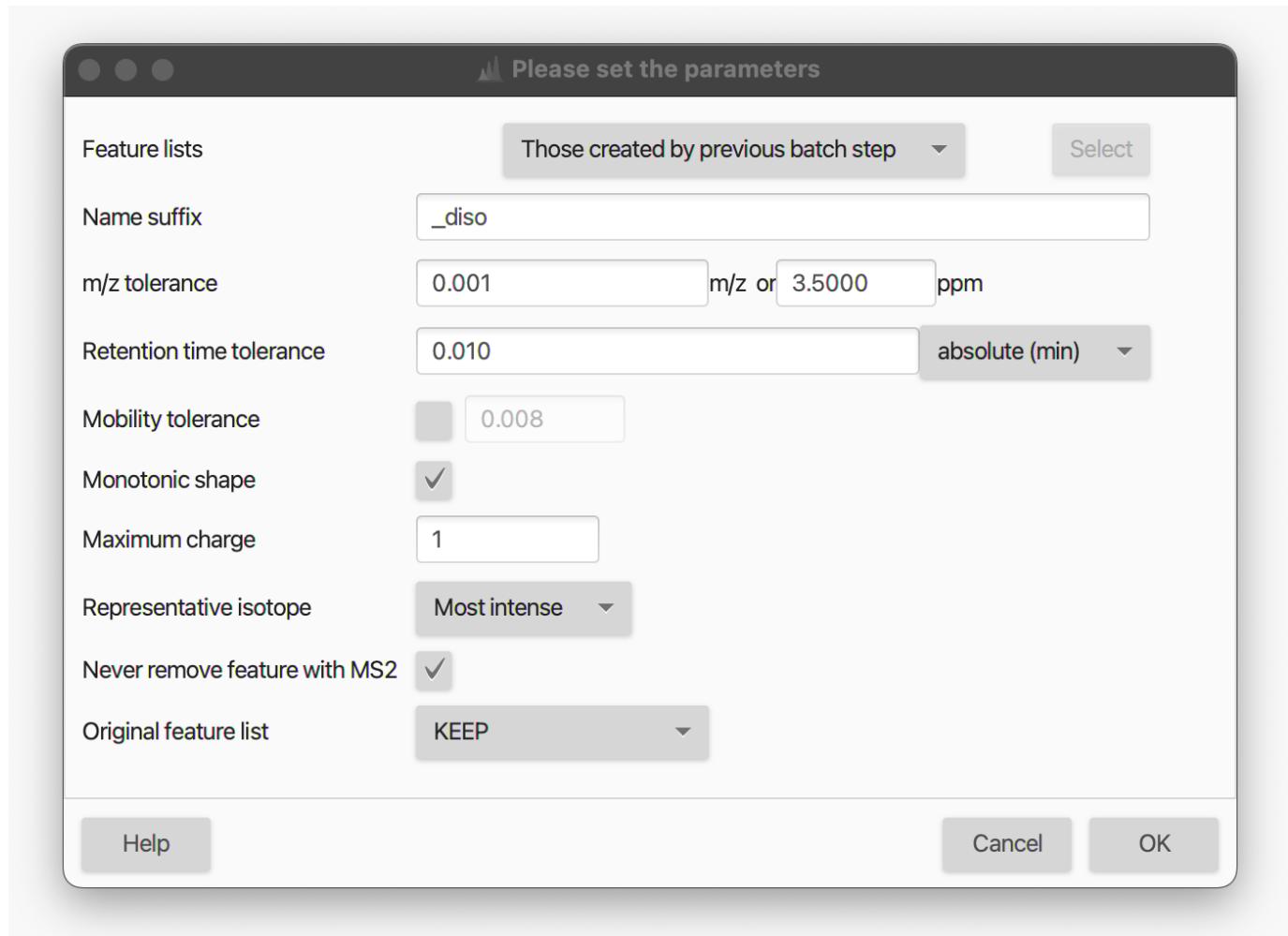
## 5.12 $^{13}\text{C}$ isotope filter

MZmine carries out the feature detection steps of *chromatogram building* and subsequent *resolving* considering all the signals stored in the *mass lists*. As a consequence, signals generated by isotopologues of the same chemical entity (see *isotopic pattern*) are detected as distinct features and included in the *feature lists*, representing redundant information for the downstream data analysis. This issue ordinarily occurs for C-containing molecules, where the  $^{13}\text{C}$  isotope signals can be easily detected ( $^{13}\text{C}/^{12}\text{C} \approx 1.1\%$ ).

The *13C isotope filter* module (formerly called *Isotope grouper*) aims at filtering out the features corresponding to the  $^{13}\text{C}$  isotopes of the same analyte. The algorithm consider each feature individually and checks for the presence of potential  $^{13}\text{C}$ -related peak(s) in the *feature lists*. When an isotope pattern meeting the user-defined tolerances (e.g.  $m/z$ , RT) and requirements (e.g. *monotonic shape*) is found, the information is saved, and only the feature corresponding to the e.g. most intense isotope is retained in the *feature list*. It must be noted that  $^{13}\text{C}$  peaks are searched within the *feature list*, and not in the raw data.

### 5.12.1 Parameters settings

:material-menu-open: Feature list methods → Isotopes → 13C isotope filter (formerly: isotope grouper)



#### NAME SUFFIX

String added as suffix when creating the new feature list(s).

**M/Z TOLERANCE**

Maximum allowed difference between the measured and the predicted  $m/z$  of the (potential)  $^{13}\text{C}$  isotope to be grouped as isotopologues. The tolerance can be specified as absolute tolerance (in  $m/z$ ), relative tolerance (in ppm), or both. When both are specified, the tolerance range is calculated using the maximum between the absolute and relative tolerances.

**💡 Tip.** We recommend to set a fairly strict  $m/z$  tolerance to reduce the risk of discarding false  $^{13}\text{C}$  isotopes.

**RETENTION TIME TOLERANCE**

Maximum allowed RT difference between the feature and its (potential)  $^{13}\text{C}$  isotope to be grouped as isotopologues.

**💡 Tip.** Isotopologues should exhibit identical chromatographic behaviour and thus produce overlapping LC peak shapes. Therefore, a strict RT tolerance can be used to reduce the risk of discarding false  $^{13}\text{C}$  isotopes.

**MOBILITY TOLERANCE**

If enabled and the mobility dimension was recorded, potential  $^{13}\text{C}$  isotopes will be grouped as isotopologues only if their mobility difference is within the defined tolerance.

**💡 Tip.** The same principle seen for the RT tolerance apply to the IM dimension. Isotopologues should exhibit identical IM separation; therefore, a strict mobility tolerance can be used to reduce the risk of discarding false  $^{13}\text{C}$  isotopes.

**MONOTONIC SHAPE**

If true, a monotonically decreasing trend of the isotope pattern (typical of the  $^{13}\text{C}$  isotope pattern of small molecules) is required for the filtering.

**MAXIMUM CHARGE**

Maximum charge state considered to predict the  $^{13}\text{C}$  isotopes'  $m/z$ . If a value  $> 1$  is set, the charge state with the maximum number of detected isotope features will be used for the filtering.

**NEVER REMOVE FEATURE WITH MS2**

If checked, potential  $^{13}\text{C}$ -related features will not be discarded if [associated to a MS2 spectrum](#).

**ORIGINAL FEATURE LISTS**

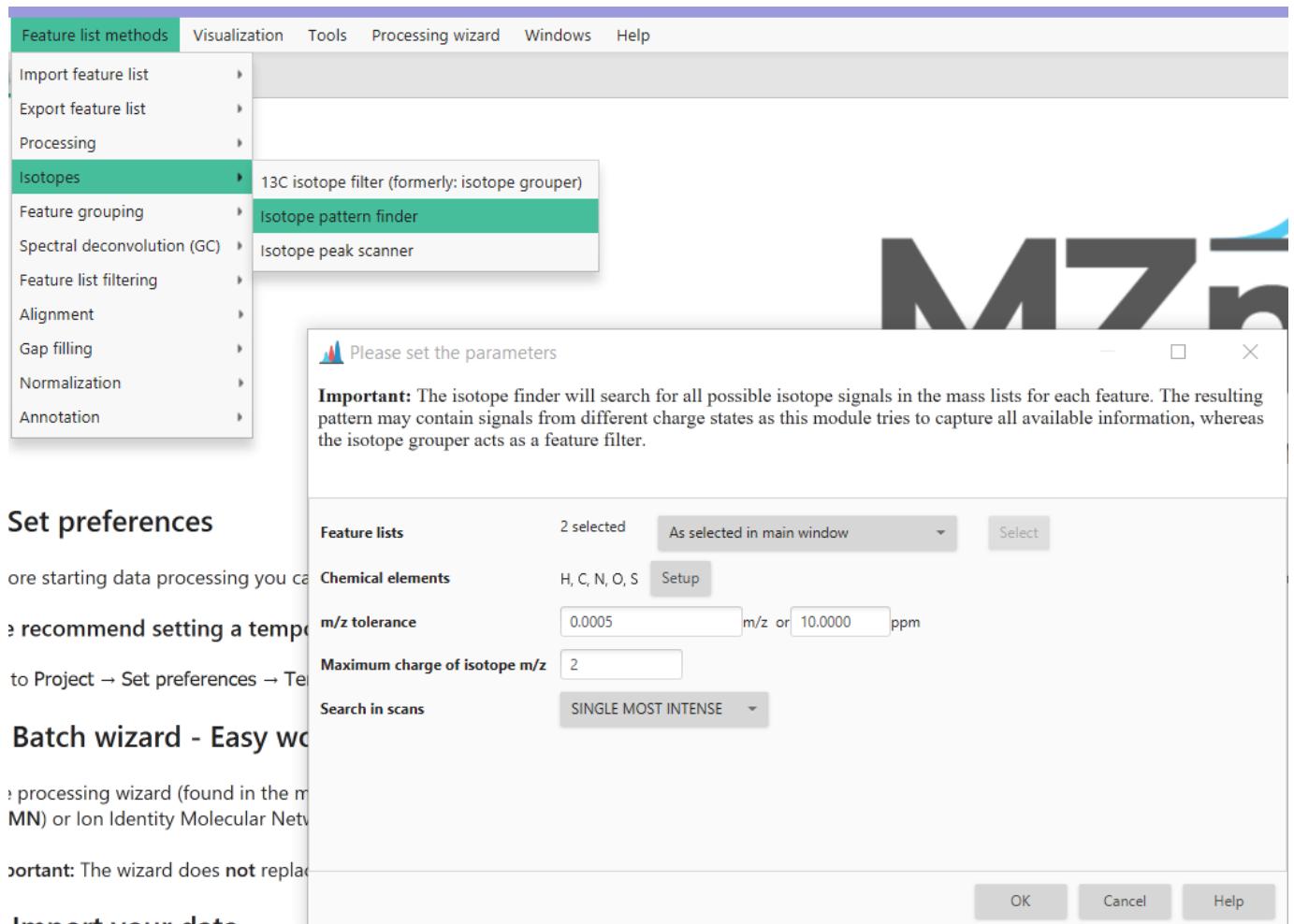
Keep or remove the input feature list(s). The *PROCESS IN PLACE* option directly filter the input feature list and performs better in terms of memory usage; therefore, it is recommended over *REMOVE*, when available.

Last update: June 23, 2022 22:22:55

## 5.13 Isotope pattern finder

The **Isotope pattern finder** is found under **Feature list methods → Isotopes → Isotope pattern finder**.

The module searches isotope patterns for each feature in selected feature lists by going back to the mass spectra. Starting from the feature m/z the algorithm will first backtrack any possible preceding isotope signals using a list of delta masses created from elements, their stable isotopes, and an m/z tolerance. For example, a -2 signal might be detected when searching for Br isotopes. In a second step, all picked potential isotope m/z values are used to search next isotope (with higher m/z). This is done for each charge state. I



### Set preferences

ore starting data processing you can

re recommend setting a temp

to Project → Set preferences → Te

**Batch wizard - Easy way**

processing wizard (found in the m

important: The wizard does not replace

**Import your data**

### Parameters

#### CHEMICAL ELEMENTS

All stable isotopes of the chosen elements are used to create a list of mass differences to search. Signals with this mass difference (m/z difference with different charge states) are then considered as potential isotope signals.

#### M/Z TOLERANCE

Maximum allowed difference between two features' m/z values in order for them to be considered the same. The value is specified both as absolute tolerance (in m/z) and relative tolerance (in ppm). The tolerance range is calculated using maximum of the absolute and relative tolerances.

**MAXIMUM CHARGE OF ISOTOPE M/Z**

Maximum possible charge of isotope m/z distributions. All present m/z values obtained by dividing isotope masses with 1,2 ...,maxCharge values will be considered. The default value is 1, but insert an integer greater than 1 if you want to consider ions of higher charge states.

**SEARCH IN SCANS**

Currently, the supported option is "Single most intense", which means the search for isotopes will happen in the single most intense MS scan of each feature.

---

Last update: June 23, 2022 23:04:07

## 5.14 Spectral library creation

---

.....  
Last update: March 22, 2022 07:51:33

## 5.15 Spectral library search

The spectral library search module can be performed on feature lists, feature list rows, or single scans. Depending on the MS level (MS1 or MS2), all corresponding query scans (e.g., extracted from the rows) will be matched against selected spectral libraries that were previously imported (formats: .json, .mgf, .msp, or .jdx).

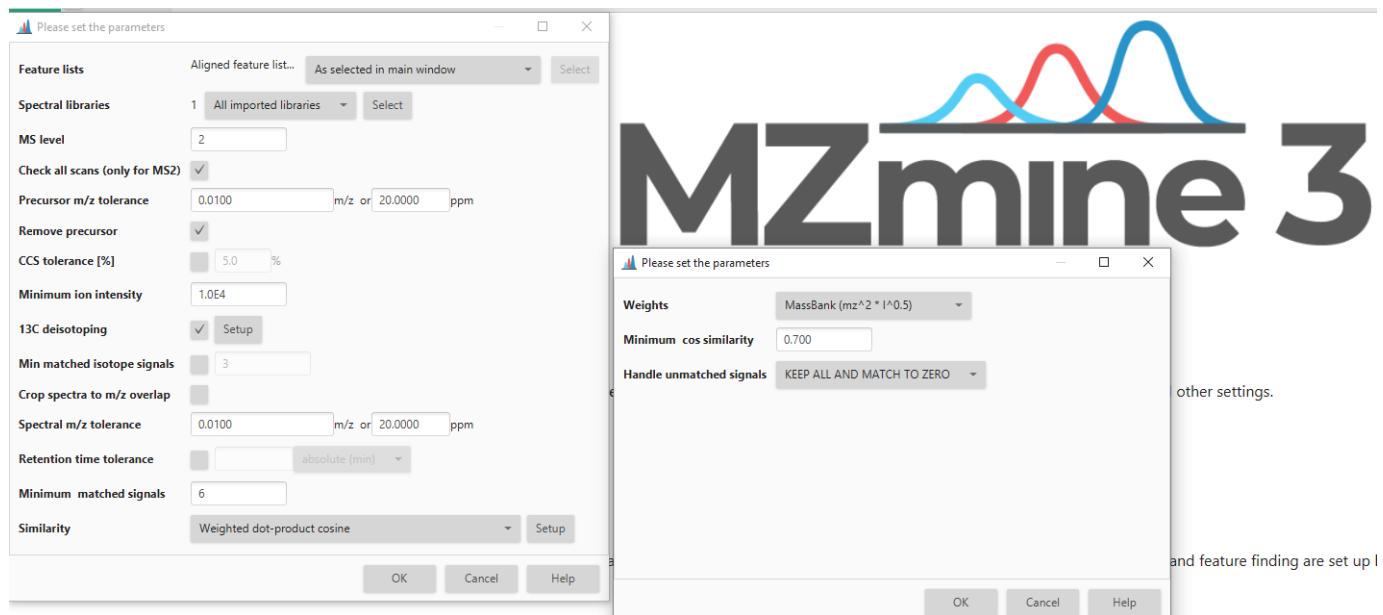
The **Spectral library search** module is found under **Feature list methods → Annotation → Search spectra → Spectral library search** or with a right click on one or multiple selected feature rows **Search → Spectral library search**

Preferred ways to import **libraries** are in this order: - together with spectral data files in the advanced data import: **Raw data methods → Raw data import → MS data (advanced)** - drag and drop into the main window - with a dedicated import module: **Raw data methods → Raw data import → Spectral library import** or under **Feature list methods → Annotation → Search spectra → Import spectral libraries**.

### 5.15.1 Supported library formats

- .json: MassBank of North America (MoNA) ([download](#))
- .json: The Global Natural Product Social Molecular Networking (GNPS) (format from the spectral DB submission module)
- .mgf: GNPS ([download](#))
- .msp: MoNA
- .msp: National Institute of Standards and Technology (NIST)
- .jdx: JCAMP-DX

#### Parameters



#### SPECTRAL LIBRARIES

The spectral libraries of interest need to be imported before applying spectral library search. Either uses all imported spectral libraries or only the selected libraries.

#### MS LEVEL

Choose the MS level of the scans that should be compared with the library. Set the MS level to "1" to compare MS<sup>1</sup> spectra, e.g., from GC-EI-MS or MALDI-imaging MS<sup>1</sup> data. Use "2" or higher for fragmentation scans. In case of issues with the scan selection, check the actual scan numbers of the scans as they are reported in the data files and in MZmine's raw data overview and scan list.

**CHECK ALL SCANS (ONLY FOR MS2)**

This option enables the comparison of all MS2 scans otherwise only the most intense MS<sup>2</sup> scan (highest TIC) is used for the matching. It does not apply to MS1 scans.

**PRECURSOR M/Z TOLERANCE**

This option is only used for MS level > 1. Here, the library entries are filtered by their precursor m/z reducing the number of spectral-pairs to match. The absolute (in m/z) and relative (in ppm) m/z tolerance can be set, whereas the maximum tolerance for each precursor is applied. Considering that the precursor isolation window is often far greater than the resolution or accuracy of the MS scan, this parameter is often set to higher m/z tolerances. Another aspect is the used library, which might contain uncalibrated reference spectra from lower resolution instruments.

**REMOVE PRECURSOR**

Can be only selected for MS level > 1. Depending on the fragmentation method, e.g., collision induced dissociation (CID) or higher-energy collisional dissociation (HCD), the precursor can be detected with different intensities resulting in varying cosine similarities during the library matching. Therefore, this option enables the removal of the precursor signal within the precursor m/z tolerance (parameter above) prior to the matching.

**SPECTRAL M/Z TOLERANCE**

This m/z tolerance is used to pair signals in the query and library scans. It can be set in absolute (in m/z) and relative (in ppm) m/z tolerance, whereas the maximum tolerance for each m/z value is applied. It must be kept in mind, which mass resolutions are achieved within the experimental spectra and within the spectral library.

**RETENTION TIME TOLERANCE**

The maximum allowed retention time difference when comparing the query and library scan. This option is useful for in-house libraries or standardized libraries that follow the same acquisition protocol with the same set up, e.g., column, instrument, and method). It can be set in absolute (min or sec) or relative (%) values.

**CCS TOLERANCE**

The collision cross-section (CCS) tolerance can be used in a similar way as the retention time tolerance. Accordingly, the CCS value of a query will be compared with the library entries and the maximum tolerance can be set in %. If the query or library entry was analyzed without ion mobility (no CCS values), no spectrum will be matched.

**CROP SPECTRA TO M/Z OVERLAP**

If query and library scans were acquired with different methods, e.g., mass range, fragmentation energy or mode, it can be helpful to crop the spectra to their overlapping m/z range (+ m/z tolerance). This is done by using the maximum m/z range where both spectra contain signals. However, this method will boost false matches and needs strict manual interpretation.

**MINIMUM ION INTENSITY**

Signals in the query scan below the minimum ion intensity will be filtered from the mass lists and are not taken into account during the library matching. Absolute values.

**<sup>13</sup>C DEISOTOPING**

Removes <sup>13</sup>C isotope signals from the mass list using the following parameters: - **m/z tolerance**: Maximum allowed difference between the measured and predicted isotope m/z values. The absolute (in m/z) and relative (in ppm) m/z tolerance can be set, whereas the maximum tolerance for each m/z value is applied. - **Monotonic shape**: If enabled, the monotonically decreasing height of isotope pattern is required. - **Maximum charge**: The maximum charge that will be considered for detecting the isotope pattern. For singly charged ions, the <sup>13</sup>C isotope will be expected +1 whereas for doubly charged ions it will be +0.5 (+1 m/z divided by the charge 2).

**MIN MATCHED ISOTOPE SIGNALS**

This option is only useful if the query AND library entries contain isotope patterns (e.g., in MS<sup>1</sup> or with wider precursor isolation windows). The minimum number of matched signals of <sup>13</sup>C isotopes. It cannot be applied when <sup>13</sup>C deisotoping is enabled.

**MIN MATCHED SIGNALS**

The query mass list and spectral library entry must contain at least this number of matched (paired) m/z values (+- m/z tolerance). This parameter must be set carefully to not exclude compounds that show less fragmentation, when using a higher number of matched signals. Choosing a lower number of matched signals can result in spurious library hits. Common parameters include 4 signals for smaller molecules and 6 for more confident matches.

**SIMILARITY**

Different algorithms can be applied to calculate the similarity of the query and library scans and to filter the resulting library matches. The most common spectral similarity measure for library search is the weighted cosine similarity.

The **weighted dot-product cosine** similarity is used for comparing MS<sup>2</sup> data, whereas the **composite dot-product identity (similar to NIST search)** considers the relative intensity of neighboring signals and is, therefore, applied to MS<sup>1</sup> spectra from GC-EI-MS.

- **Weights:** For calculating the cosine similarity, different weighting strategies for m/z and signal intensities can be applied. A usual scaling is based on the square root of the peak intensity (sometimes multiplying by the squared m/z to value larger fragments over smaller ones).
- **Minimum cosine similarity:** This option defines the minimum accepted similarity score that is taken into account for annotation. The similarity score depends on the data handling of unmatched signals.
- **Handle unmatched signals:** Signals that only occur within one scan (query OR library entry) can be handled in different ways to affect the cosine similarity and controlling the quality of matching results.
- **Keep all and match to zero** This is the **standard** conservative approach where all unmatched signals weigh negatively on the score. It is used for both GC-EI-MS and MS<sup>2</sup> spectra.
- **Remove all** is the opposite option and discard all unaligned signals, which increases the similarity score artificially. This option is only feasible if both the library and query spectrum are considered to be complex mixtures. Therefore, the last two options are more conservative.
- **Keep library signals** results in discarding all unaligned signals of the query scan, whereas all unaligned library signals are matched to zero, setting the library spectrum as the ground truth. Here, the negative impact of contaminating signals in the query scans are reduced. This might be helpful for mixed spectra of multiple compounds, especially during imaging techniques without any further separation or all ion fragmentation/data independent fragmentation workflows.
- **Keep experimental signals** results in discarding all unaligned signals of the library scan, whereas all unaligned query signals are matched to zero. With reversed effects compared to the previous option.

Last update: June 23, 2022 23:17:49

## Recommended Citations

**IIMN:** Schmid R., Petras D., Nothias LF, et al. [Ion Identity Molecular Networking for mass spectrometry-based metabolomics in the GNPS Environment](#). Nat. Comm. 12, 3832 (2021).

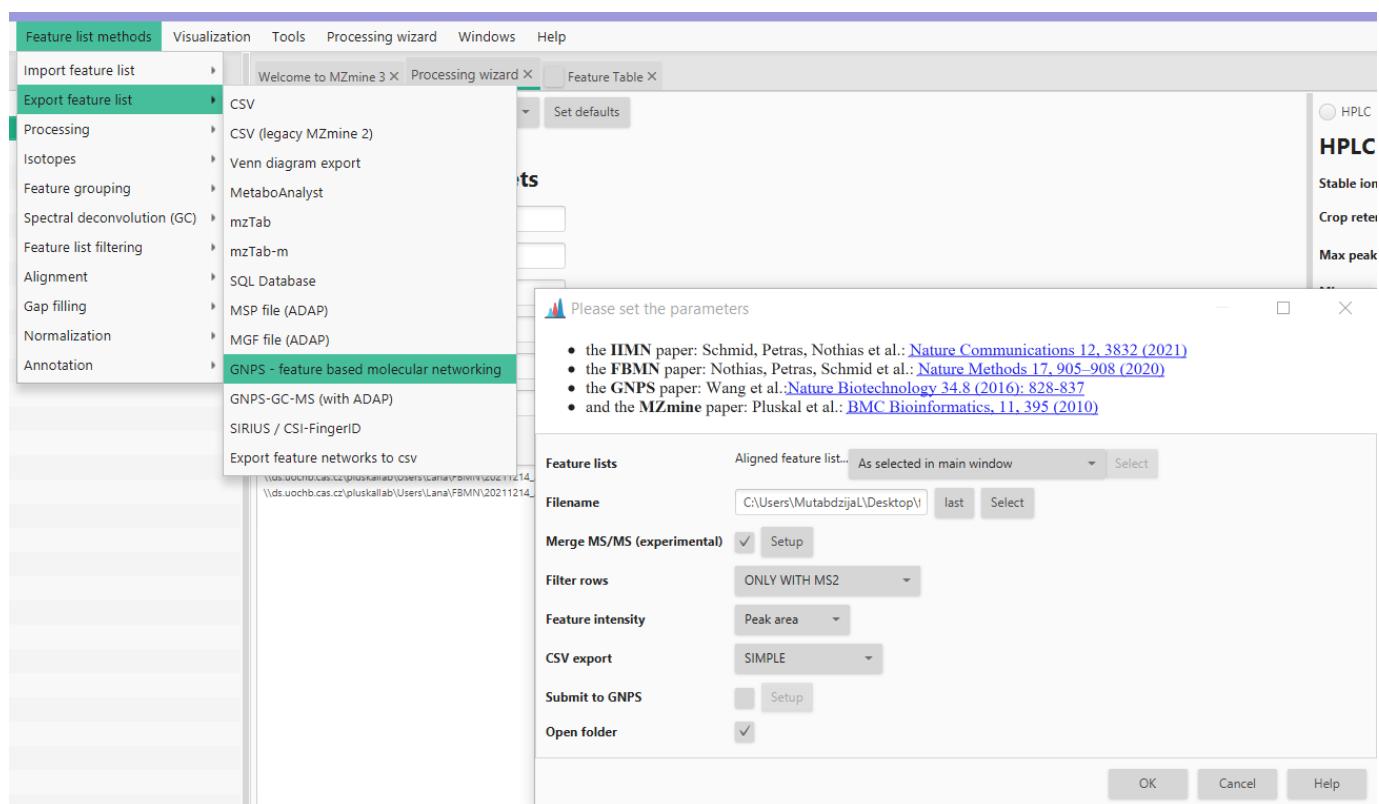
**FBMN:** Nothias, L.-F., Petras, D., Schmid, R. et al. [Feature-based molecular networking in the GNPS analysis environment](#). Nat. Methods 17, 905–908 (2020).

**GNPS:** Wang, M. et al. [Sharing and community curation of mass spectrometry data with Global Natural Products Social Molecular Networking](#). Nat. Biotechnol. 34, 828–837 (2016).

## 5.16 GNPS-FBMN/IIMN export

This module connects MZmine feature finding results to the [GNPS](#) workflows for [Feature-based Molecular Networking \(FBMN\)](#) and [Ion Identity Molecular Networking \(IIMN\)](#).

The export module is found under **Feature list methods → Export feature lists → GNPS - feature based molecular networking**.



Using this module, the user can export the feature list needed for the manual submission to GNPS' feature based molecular networking (GNPS FBMN) or directly submit the job to the GNPS platform from MZmine. In both cases, two files are created:

1. Quantification table (CSV file) which contains the features and their associated information (e.g., average m/z, retention time, and each feature's area or height).
2. MS/MS spectral summary (.MGF file) which contains one representative MS/MS spectrum for each row in the feature list.
3. A [supplementary edges file](#) with related ion identities (if ion identity networking was performed).

## Parameters

### FILENAME

Name to be given to the output files (.MGF and .CSV). In this field, the user can either write the path where they want to save the file, or click "select", navigate into the desired output folder, write the output name in the "file name" field and click save. Once that is done, the path should be visible in the Filename field in the GNPS export module.

### MERGE MS/MS (EXPERIMENTAL)

If checked, high quality MS/MS spectra that correspond to one feature are merged, instead of exporting only the most intense MS/MS spectrum. See [MS/MS merger](#) for additional information.

### FILTER ROWS

In the final output files, the user can select to export all the rows without any filters applied, rows only with MS/MS spectra, rows with MS/MS and Ion Identity (it gives MS/MS and the adduct information) and rows with MS/MS or Ion Identity. Normally, for FBMN you want to retain features with MS/MS spectra.

### FEATURE INTENSITY

The user can either select peak area or peak height which will then be displayed in the quantification table.

### CSV EXPORT

The user can choose between **simple**, **comprehensive**, or **all**. Difference is in the amount of information that is present in the quantification table. Simple resembles the legacy format from the MZmine 2 export. Both options can be used for FBMN in GNPS other tools might rely on the simple MZmine 2 style output.

### SUBMIT TO GNPS

This option allows any user to directly submit FBMN/IIMN jobs to GNPS. The password and user name are optional and are sent without encryption (until the server has moved to its final location with https). The input files uploaded to GNPS with the "Submit to GNPS" option are not saved on your GNPS user account. These files are deleted on monthly basis, which prevent future cloning of the job and retrieval of the files. Use the "standard" interface of FBMN for persistant jobs and more options. Or log into your GNPS account and click on **Clone to latest version** for a job submitted via direct interface.

### OPEN FOLDER

Opens the export folder.

Last update: April 14, 2022 00:14:12

## 5.17 Join aligner

This method aligns detected peaks in different samples through a match score. This score is calculated based on the mass and retention time of each peak and ranges of tolerance stipulated in the parameter setup dialog.

The join aligner module is found under **Feature list methods → Alignment → Join aligner**.

Please set the parameters

Feature lists	Aligned feature list	As selected in main window	Select
Feature list name	Aligned feature list		
m/z tolerance	0.0010	m/z or	5.0000 ppm
Weight for m/z	3		
Retention time tolerance	0.100	absolute (min)	
Weight for RT	1		
Mobility tolerance	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Mobility weight	<input type="checkbox"/>		
Require same charge state	<input type="checkbox"/>		
Require same ID	<input type="checkbox"/>		
Compare isotope pattern	<input type="checkbox"/>	Setup	
Compare spectra similarity	<input type="checkbox"/>	Setup	
Original feature list	KEEP		
<b>OK</b> <b>Cancel</b> <b>Help</b>			

**ALGORITHM**

The peak alignment algorithm uses:

1. a master list of peaks (L) against which every new sample ( $S_j$ ) will be matched. When aligning peaks from multiple samples, the master list is initially set to the first sample, and subsequently it will be the combination of samples aligned thus far, with the samples as the columns and the matching peaks as the rows.
2. for every row  $i$  in L, a two-dimensional window (where the window size is selected by the user), called Alignment window (AW) defining the ranges of m/z and RT and centered around the average of m/z and RT of all the individual peaks in the row; and
3. a score function to compute the similarity of peaks between L and the new sample  $S_j$  inside the alignment window. The score function computes the similarity based on the similarities in m/z, retention time, and optionally on identification, and isotope patterns between the peaks to be matched. The score is calculated as follows:  $\text{score} = ((1 - \text{MZdifference} / \text{MZtolerance}) \times \text{MZweight}) + ((1 - \text{RTdifference} / \text{RTtolerance}) \times \text{RTweight})$

The algorithm works as follows. It iterates through the rows of L, and for each row, it looks for peaks within the alignment window in  $S_j$  that has to be aligned with L. A score is calculated for each possible match and the pair getting the best score will be aligned.

**Parameters****FEATURE LIST NAME**

Name of the new aligned peak list.

**M/Z TOLERANCE**

Maximum allowed difference between two m/z values in order for them to be considered the same and thus the peaks aligned. The value is specified both as absolute tolerance (in m/z) and relative tolerance (in ppm). The tolerance range is calculated using maximum of the absolute and relative tolerances for possible peaks to be aligned.

**WEIGHT FOR M/Z**

This is the assigned weight for m/z difference at the moment of match score calculation between peak rows, as can be seen in the aforementioned formula. Only in cases where there is a perfect match of m/z values, the score receives the complete m/z weight. Generally, higher weight is given to m/z values than to RT values.

**RETENTION TIME TOLERANCE**

Maximum allowed difference between two retention times in order for them to be considered the same and thus peaks aligned. Maximum RT difference can be defined either using absolute or relative value.

**WEIGHT FOR RT**

This is the assigned weight for RT difference at the moment of match score calculation between peak rows. Only in cases where there is a perfect match of RT values, the score receives the complete RT weight.

**MOBILITY TOLERANCE**

In case of IM data, the user can determine the mobility tolerance. If checked, this parameter specifies the tolerance range for matching the mobility values.

**MOBILITY WEIGHT**

Score for perfectly matching mobility values. Only taken into account if "Mobility tolerance" is activated. Furthermore, score calculation that is mentioned in the **Algorithm** is then modified to account for the mobility as well. Mobility tolerance and weight are accounted for in the same manner as m/z and RT parameters.

**REQUIRE SAME CHARGE STATE**

If checked, only rows having same charge can be aligned.

**REQUIRE SAME ID**

If checked, only rows having same compound identities (or no identities) can be aligned.

#### COMPARE ISOTOPE PATTERN

If both peaks represent an isotope pattern, checking this box will add isotope pattern score to the match score calculation. Additionally, the user can set up **isotope m/z tolerance** which defines what isotopes would be considered same when comparing two isotopic patterns, **minimum absolute intensity** below which isotopes will be ignored and **minimum score %** between isotope patterns that has to be satisfied in order for the match to not be discarded.

#### COMPARE SPECTRA SIMILARITY

Compare MS1 or MS2 scans similarity. Select the m/z tolerance, MS level and spectra similarity algorithm. Only features meeting this criteria will be aligned. See [compare spectra similarity](#) for additional information.

#### ORIGINAL FEATURE LIST

The user can choose to either KEEP the original feature list and generate a new processed one, or REMOVE the original feature list with the processed one. Generally, you would keep the original feature list, but opting for REMOVE will save memory.

---

Last update: May 6, 2022 09:41:02

## 5.18 Other parameters

### 5.18.1 Merge MS/MS (experimental)

This option is available in the [GNPS FBMN export](#) and the SIRIUS export. If checked, high quality MS/MS spectra that correspond to one feature are merged, instead of exporting only the most intense MS/MS spectrum.

#### PARAMETERS

Please set the parameters

Select spectra to merge	across samples
m/z merge mode	weighted average (remove outliers)
intensity merge mode	sum intensities
Expected mass deviation	0.0010 m/z or 10.0000 ppm
Cosine threshold (%)	70.0 %
Signal count threshold (%)	20.0 %
Isolation window offset (m/z)	0.0000
Isolation window width (m/z)	1.0000

#### Select spectra to merge

The users can select to merge the MS/MS spectra:

1. **Across samples**, which will merge all MS/MS spectra that belong to the same feature, and as such is the most convenient option.
2. **Same sample**, which will merge MS/MS spectra for the same feature within one sample, and can be used if the user is not confident about the alignment algorithm.
3. **Consecutive scans**, which will merge MS/MS spectra if they are triggered in a row.

#### m/z merge mode

This option allows you to select the way to merge the fragments' m/z values associated with a similar precursor value. "Most intense" will always pick the m/z of the best feature, which is a very safe and conservative option. However, "weighted average (remove outliers)" will often yield better result.

**Intensity merge mode**

Options on how to merge the intensity values of features from different spectra with similar mass.

- **Sum intensities** is a convenient option that will increase the intensities of feature that occur consistently in many fragment scans. However, this will make intensities between merged and unmerged spectra incomparable.
- Use **max intensity** to preserve intensity values

**Expected mass deviation**

Expected mass deviation between different spectra of the same feature of your measurement in ppm (parts per million) or Da(larger value is used). We recommend to use a rather large values, e.g. 10ppm for Orbitrap, 15 ppm for Q-ToF, 100 ppm for QQQ.

**Cosine treshold**

Treshold of cosine similarity between spectra that needs to be met in order for two spectra to be merged. In case they have different collision energies, cosine treshold should be set to 0, since different collision energies will result in different fragmentation pattern.

**Signal count treshold**

After merging the spectra, signals that occur in less than the user specified % of the merged spectra will be removed.

**Isolation window offset (m/z)**

Isolation window offset from the precursor m/z.

**Isolation window width (m/z)**

Width of the isolation window (left and right).

---

Last update: April 14, 2022 00:14:12

## 5.18.2 Spectra similarity

### Spectral m/z tolerance

Spectral m/z tolerance is used to match all signals between spectra of two compared raw files.

### MS level

MS level of scans that should be compared. It can be 1 for MS1 or 2 for MS2 level.

### Compare spectra similarity

1. **Weighted dot-product cosine** - used to determine the similarity between two spectra (usually library and query spectra). This option is used for MS2 level.
2. **Composite dot-product identity** - used to determine the similarity between two spectra (usually library and query spectra). Especially useful for very reproducible generation of spectra (GC-EI-MS). Takes into account the relative intensities of neighbouring signals in the two spectra. This option is used for MS1 level.

Additional setup of spectra similarity comparison enables modification of the following parameters:

### Weights

Weights for the m/z and intensity values. Usually, MassBank is used, in which higher m/z values contribute more to the cosine similarity calculation.

### Minimum cos similarity

Minimum cosine similarity for a match between compared spectra.

### Handle unmatched signals

Usually, **keep all and match to zero** is used, which will take all signals into account, and the unmatched ones will decrease the cosine similarity.

---

Last update: April 29, 2022 14:53:51

## 6. Workflows

---

### 6.1 LC-MS Workflow

The workflow proposed herein is intended as a general pipeline for untargeted LC-MS (or LC-MS/MS) data preprocessing. The main goal is essentially to turn the highly-complex LC-MS raw data into a list of features, and corresponding signal intensity, detected across the analysed samples. Such feature lists can then be exported for further downstream analysis (e.g., identification, search against spectral libraries, statistical analysis, etc.). A schematic representation of the workflow is shown below:



References: - 10.1039/9781788019880-00232 - 10.1007/978-1-0716-0239-3\_3 - 10.1016/bs.coac.2018.08.003

#### 6.1.1 Raw data processing

The raw data processing consists of essentially two steps: [Data import](#) and [Mass detection](#)

##### Raw data import

Either open (e.g. mzML) and native vendor (e.g. Thermo, Bruker) data formats can be imported in MZmine 3. All the supported formats are listed here ([LINK to Doc](#)). For more details see the [Data import](#) module.

##### Mass detection

This step produces a list (referred to as "mass list") of the m/z values found in each MS scan across the LC run that exceed a user-defined threshold (i.e. noise level). For more details see the [Mass detection](#) module.

#### 6.1.2 Feature processing

The goal of the "Feature processing" is to obtain a list of all the detected features (characterized by a RT and m/z value) from the raw LC-MS data.

##### Chromatogram building

The first step in the "Feature processing" is to build the so-called extracted ion chromatograms (EICs) for each detected mass (see "Mass detection"). There are two modules in MZmine 3 that can fulfil this task: [ADAP chromatogram builder](#) (widely used) and [Grid mass](#) (create docs).

The "detected" features in each file are listed in the so-called "feature lists", which are then further processed.

(e.g. to ) and aligned to connect corresponding features across all samples.

##### Smoothing in retention time dimension (optional)

- Optional, depends on the LC peak shape
- For more details see the [Mass detection](#) module.

##### [Smoothing](#)

## Feature resolving

### Local minimum resolver

#### [13C isotope filter \(isotope grouper\)](#)

- Removes  $^{13}\text{C}$  isotope features from the feature list
- Has limited chance to detect isotope patterns for all features as only detected features (with all their filters and constraints are considered)
- Use the isotope finder for more sensitive detection of possible isotope signals

#### [13C isotope filter \(isotope grouper\)](#)

## Isotope pattern finder

- Searches for the isotope signals of selected chemical elements in the mass list of each feature.
- The isotope pattern detected by the **isotope finder** module has priority over the one detected by the **isotope filter (grouper)** module, if both are available.

### Isotope pattern finder

## Join aligner

- Aligns detected peaks in different samples through a match score. The score is calculated based on the mass and retention time of each peak and ranges of tolerance stipulated in the parameter setup dialog. For more information, see the [join aligner](#) module.

## Gap-filling

Gap-filling can be performed on the aligned feature lists to cope with missing features that might be artifacts of the feature-detection process. - For more details see the Gap-filling module.

---

## 6.1.3 Page Contributors

[SteffenHeu](#) (44.29%), [tdamiani](#) (25.71%), [Robin Schmid](#) (2.86%), [lalalana5](#) (18.57%), [lalalana5](#) (8.57%)

---

Last update: April 29, 2022 15:01:43

## 6.2 LC-IMS-MS Workflow Overview

---

Compared to regular LC-MS, LC-IM-MS data is more complex due to the additional separation dimension. Since some terms might not be straightforward for new users, a basic explanation of IM separation principles and the terminology used within this documentation is provided [here](#).

### 6.2.1 Supported formats

---

- Vendor formats:
  - .tdf (Native Bruker LC-IMS-MS and MALDI-IMS-MSI format)
  - .tsf (Native Bruker MALDI-IMS-MS (single shot) format)
  - .mzML
  - Created via [MSConvert](#) from native Bruker data
  - Created via [MSConvert](#) from native Waters/Agilent data
- 

### 6.2.2 Feature detection workflows

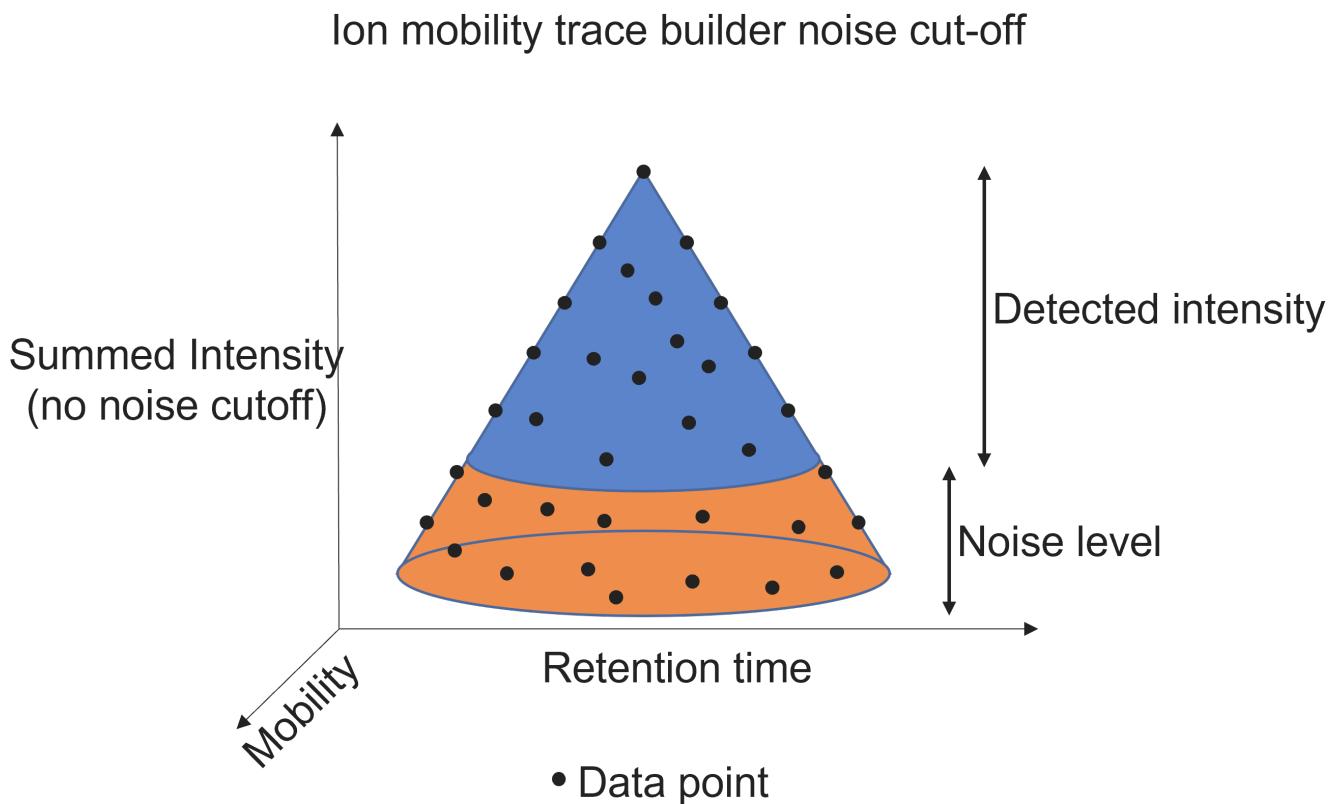
---

Ion mobility data can be processed in MZmine 3 in two ways. The first few steps are different for the two workflows (see below).

1. [LC-IMS-MS workflow via ADAP Chromatogram builder and IMS expander \( recommended\)](#)
2. [LC-IMS-MS workflow via Ion mobility trace builder / Recursive IMS builder](#)

While these lists might seem fairly similar, there are some differences in the processing approach. The LC-IMS-MS workflow builds ion mobility traces from the data in the mobility scans, whilst the LC-MS workflow builds EICs from the summed frames. For ion mobility data imported from .mzML files, accumulated frame spectra have to be built from the individual mobility scans after [mass detection](#). Since the mass detection impacts the computation of accumulated frame spectra in the same way it would impact the [ion mobility trace builder](#), the differences from this workflow and the [ADAP workflow](#) will be negligible. However, frame spectra for native Bruker .tdf raw data are summed by the vendor library during file import. Here, the frame spectra are generated from the raw data and thus result in higher intensities, since the low abundant data points on the edges of

the mobility and retention time peaks are not cut-off by the mass detection step. (see below)



Therefore, the more low abundant compounds might be detected, if the LC-MS workflow is recommended.

#### LC-MS workflow (recommended)

LC-IMS-MS data can also be processed via the regular LC-MS modules. If necessary, detected features can be expanded into the mobility dimension.

For this workflow, generation of summed frame spectra via the [Mobility scan merging](#) module is a mandatory step, if the data was imported from an .mzML file (automatically generated via native Bruker import).

- [Data import](#)
- [Mass detection](#)
- [Mobility scan merging \(mzML data\)](#)
- [ADAP Chromatogram builder](#)
- [Smoothing in retention time dimension \(optional\)](#)
- [Resolving in retention time dimension](#)
- [Expanding EICs in mobility dimension](#)
- [Smoothing in mobility dimension \(optional\)](#)
- [Resolving in mobility dimension](#)
- [Smoothing in rt and mobility dimension \(optional\)](#)
- Some recognised features might have rather noisy signals (in rt and mobility dimension) after the mobility resolving step. If smoother shapes are required, the smoothing can be reapplied afterwards. In that case, smoothing can be applied to both dimensions at once.

### LC-IMS-MS workflow

The LC-IMS-MS workflow will directly build [ion mobility traces](#) from the raw data in the mobility scans. This workflow does not necessarily require summed frame spectra. However, if extracted ion chromatograms shall be visualized via the [Chromatogram visualizer](#), the frame intensities are used. In case these are not present, the chromatograms will be blank. Note that feature intensities from the LC-IMS-MS workflow might not exactly match the frame chromatograms due to summing being executed prior to thresholding (for native Bruker data). Furthermore, multiple isomers might hide behind a single chromatographic peak.

- [Data import](#)
- [Mass detection](#)
- [Ion mobility trace builder](#)
- [Smoothing in retention time dimension \(optional\)](#)
- [Resolving in retention time dimension](#)
- [Smoothing in mobility dimension \(optional\)](#)
- [Resolving in mobility dimension](#)
- [Smoothing in rt and mobility dimension \(optional\)](#)
- Some recognised features might have rather noisy signals (in rt and mobility dimension) after the mobility resolving step. If smoother shapes are required, the smoothing can be reapplied afterwards. In that case, smoothing can be applied to both dimensions at once.

### 6.2.3 Graphical comparison of LC-MS and LC-IMS-MS data

#### Data comparison

### 6.2.4 Page Contributors

[SteffenHeu](#) (94.12%), [tdamiani](#) (4.9%), [Olena Mokshyna](#) (0.98%)

---

Last update: July 22, 2022 13:14:57

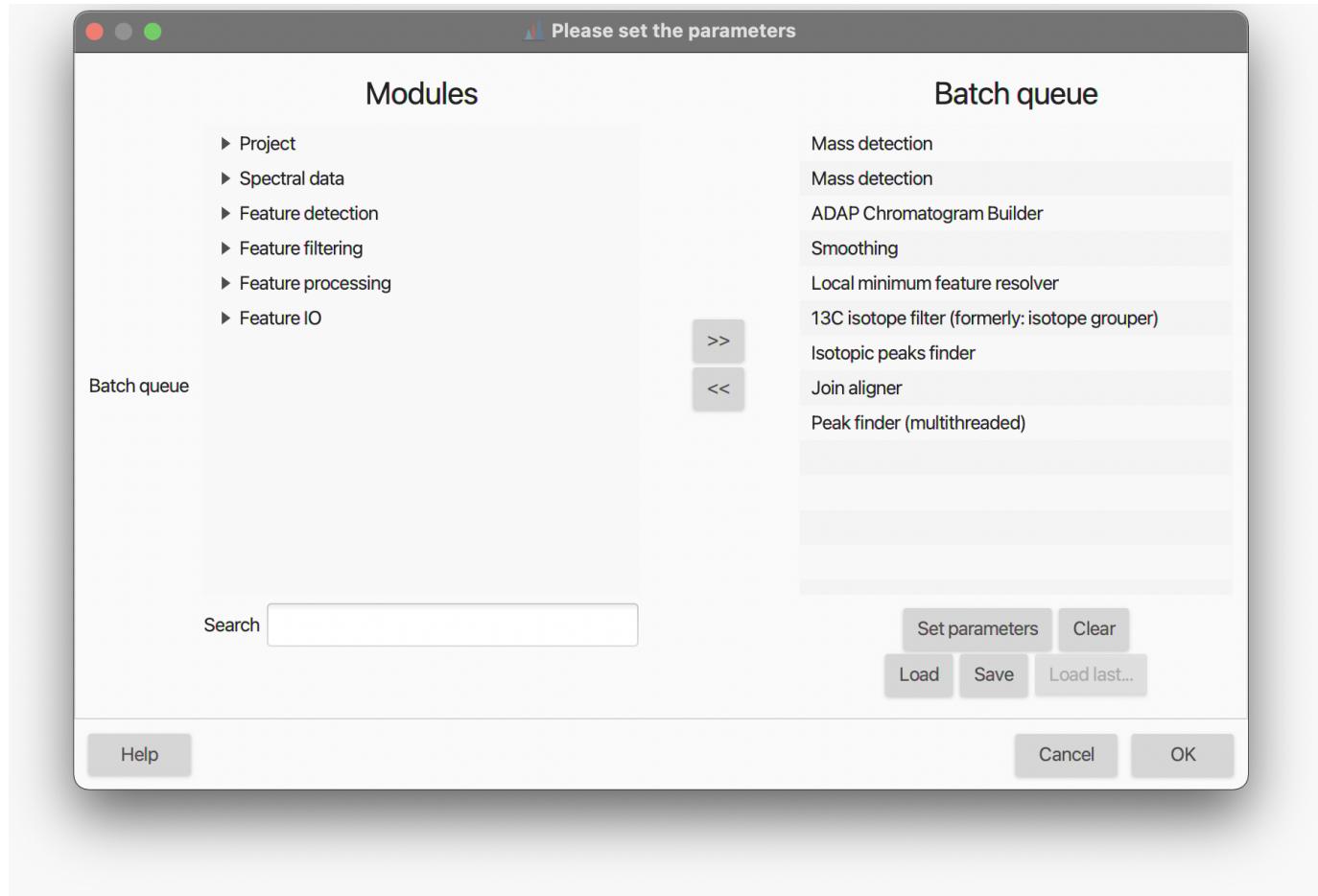
## 6.3 Batch processing

Besides the interactive [GUI](#), MZmine allows the user to run processing workflows in an automated manner using the "batch mode". Entire processing pipelines (including data import/export) can be run with few clicks, or even through the command-line application. This makes MZmine suitable to be integrated into automated data analysis pipelines (e.g. QC systems).

Batch files (XML format) are essentially lists of tasks run by MZmine one after another. Any of the methods available in MZmine 3 can be included in the batch file.

### 6.3.1 How to run batch processing

#### Project :material-arrow-right-thin: Batch mode



When a new step is added to the queue its parameter setup dialog is shown. The "Set parameters" button allows the user to modify a step's parameter settings. The "Clear" button removes all steps. The "Load" and "Save" buttons make it possible to read and write batch steps to XML files.

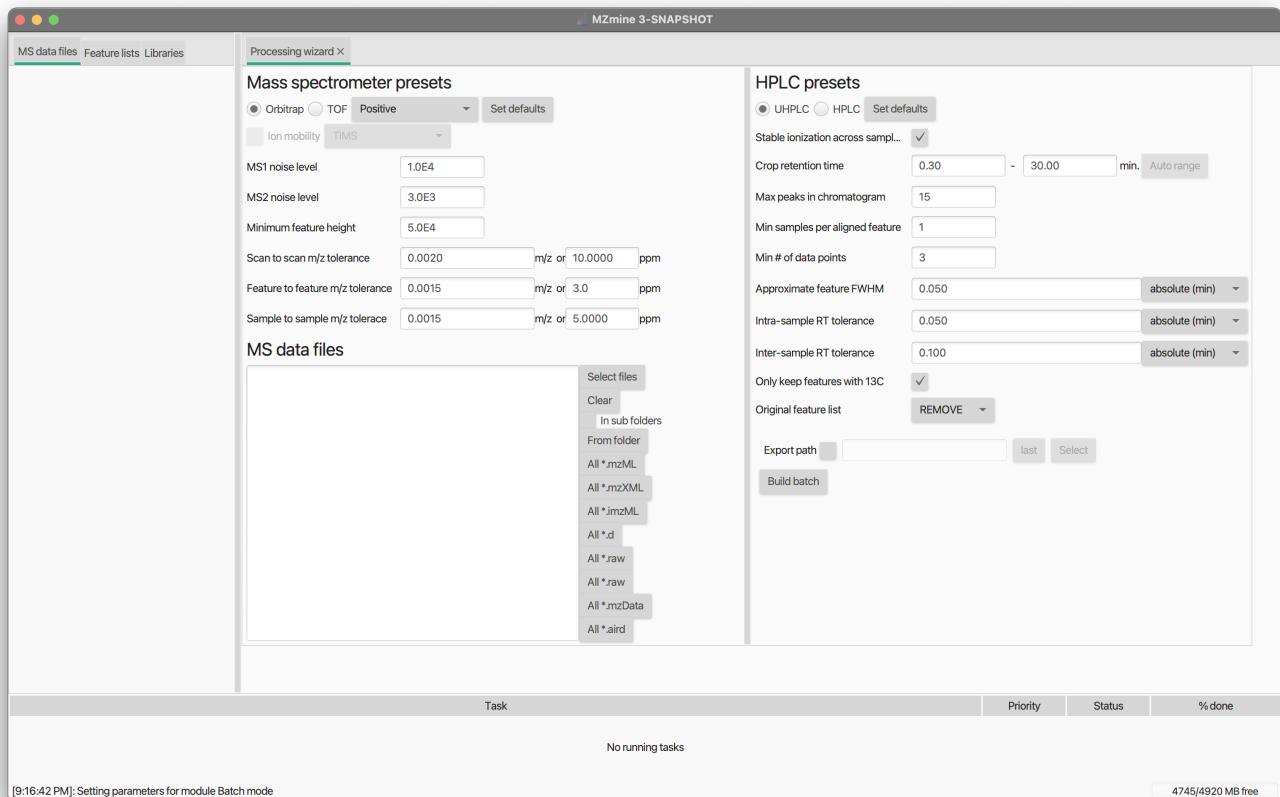
The first step of a batch queue is performed on those raw data files and/or peak lists selected by the user. The remaining steps are performed on the results produced by each preceding step (File/Feature list selection must be set to *Those created by previous batch step*). For example, if the first step of the batch queue is the [ADAP chromatogram builder](#), it will produce peak lists as a result. If the following step were Peak list deconvolution then it will be performed on the peak lists produced by the preceding Chromatogram builder step.

:octicons-alert-16: **Tip** MZmine "remembers" the last settings used.

## 6.4 Processing wizard

The processing wizard is intended to quickly set up a general workflow for the processing of untargeted LC-MS and LC-IM-MS data. By clicking the "Set default" button, default settings for mass and feature detection are also provided according to the selected MS type (Orbitrap or TOF) and LC system (UHPLC or HPLC). Once the desired parameters have been set, hit the "Build batch" button and a pre-populated batch window will open up.

### Tools :material-arrow-right-thin: Processing wizard



#### Mass spectrometers presets

**MS type:** When TOF is selected, the "Ion mobility" can be enabled

**MS1 and MS2 noise level:**

**Minimum feature height:**

**Scan to scan m/z tolerance:**

**Feature to feature m/z tolerance:**

#### HPLC presets

**Stable ionization across samples:**

**Crop retention time:**

**Max peaks in chromatogram:**

**Min samples per aligned feature:**

**Min # of data points:**

**Approximate feature FWHM:**

**Intra-sample RT tolerance:**

**Inter-sample RT tolerance:**

**Only keep feature with 13C:**

Original feature list:

Export path:



The default settings were optimized on sample datasets used during the MZmine 3 development. Although probably suitable for many applications, it is strongly recommended not to blindly rely on them. Rather, optimal processing parameters should be chosen based on the LC-(IM)-MS system performance and data acquisition settings.

---

Last update: April 5, 2022 19:53:57

## 7. Additional resources

---

### 7.1 General terminology

#### 7.1.1 MS

##### Precursor and fragment ions

The precursor ion (a.k.a. "parent ion") is the ion that dissociates to a smaller fragment ions in a MS/MS experiment. A fragment ion (a.k.a. "daughter ion" or "product ion") is the charged product of an ion dissociation. A fragment ion may be stable itself or may dissociate further to form other charged fragment ions and neutral species of successively lower mass.

##### Accurate mass, exact mass and mass accuracy

The accurate mass is the experimentally-determined mass of an ion measured with an high-resolution mass spectrometer. The exact mass is the calculated mass of an ion based on its elemental formula, isotopic composition and charge state. While the accurate mass is an experimentally-measured quantity, the exact mass is a theoretically-calculated quantity. The mass accuracy is defined as the difference between the measured value (accurate mass) and the true value (exact mass). It can be expressed either in absolute (mDa) or relative (ppm) terms.

##### Monoisotopic mass

Exact mass of an ion calculated using the mass of the lightest isotope of each element.

##### Isotopic pattern

To-do

##### Mass resolution

Todo. Often called/expressed as mass resolvin power

##### Full scan acquisition mode

Todo

##### Data-dependent acquisition mode

In data-dependent acquisition (DDA) schemes, the mass spectrometer detects 'suitable' precursor ions in each MS scan and selects them for fragmentation in consecutive MS2 scans.

Todo:'Cycle time' and 'topN' acquisition schemes

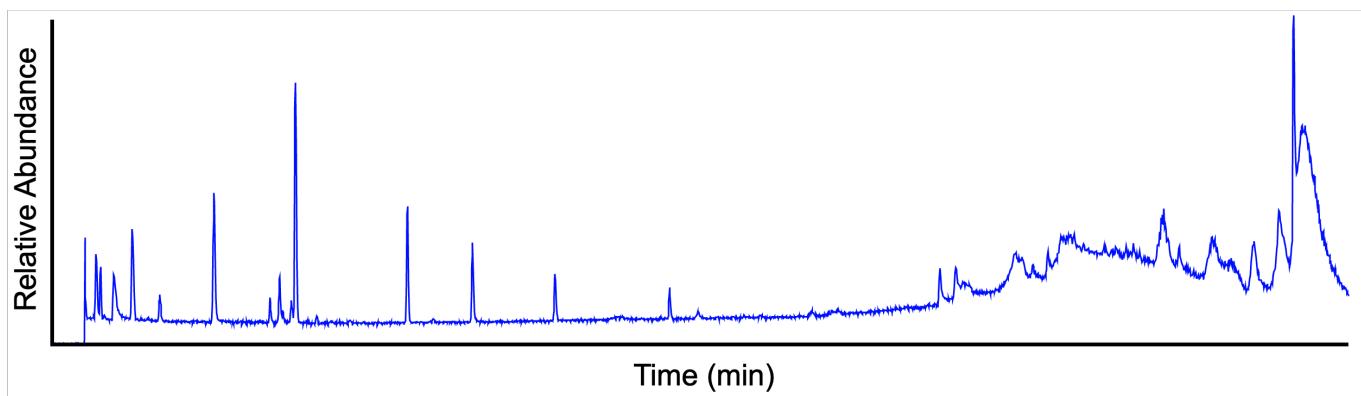
##### Data-independent acquisition mode

Todo

#### 7.1.2 LC-MS

##### Total ion current chromatogram

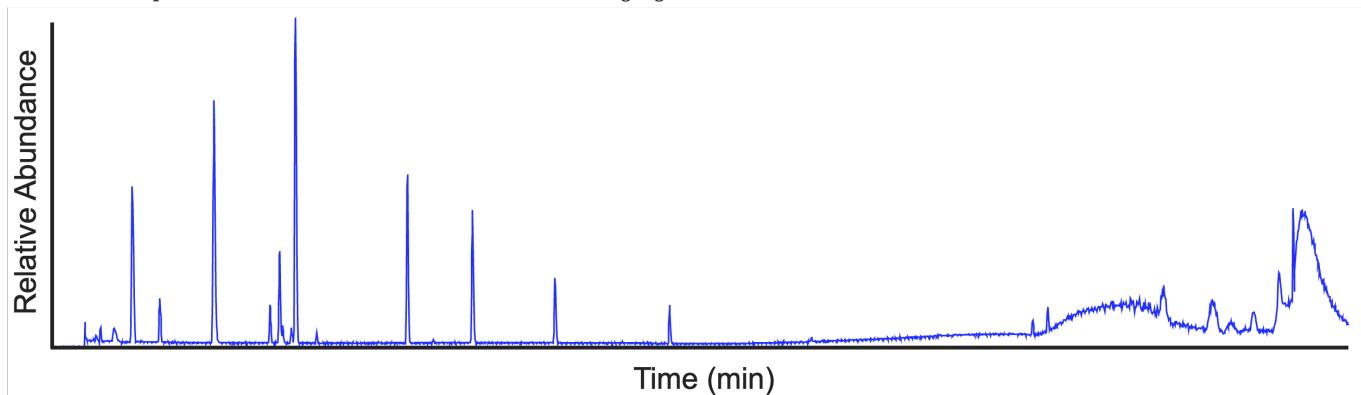
The total ion current (TIC) chromatogram displays the summed signal intensity (y-axis) over the entire m/z range at any one retention time point (x-axis) in the LC-MS run. The following figure shows a TIC chromatogram of a 9-compounds mixture analysed on LC-MS system.



*Note.* In complex samples, the TIC chromatogram often provides limited information as multiple analytes elute simultaneously, obscuring individual species.

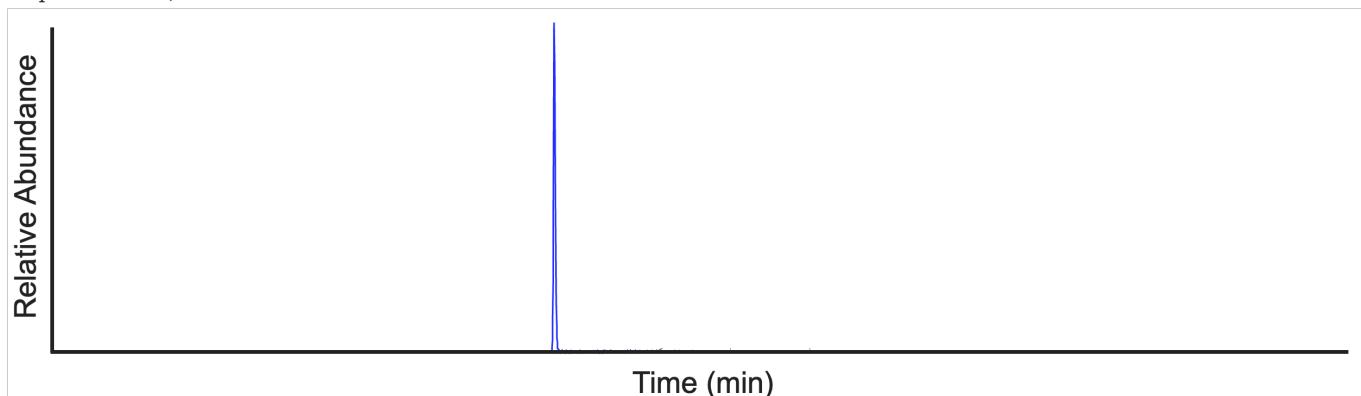
#### Base peak chromatogram

The base peak chromatogram (BPC) displays the signal intensity of the most intense mass peak in the MS spectra at any one retention time point (x-axis) in the LC-MS run. The following figure shows the same data as above, visualized in BPI mode.



#### Extracted ion chromatogram

The extracted ion chromatogram (EIC) displays the signal intensity of a specific  $m/z$  value, within a defined tolerance (e.g.  $\pm 5$  ppm), at any one retention time point in the LC-MS run. The following figure shows the EIC of  $m/z$   $455.2945 \pm 5$  ppm (same sample as above).



#### Chromatographic resolving

Peak overlapping, or co-elution, is a common problem in any chromatographic separation technique. In the case of LC-MS (especially untargeted *omics* analysis), it is virtually impossible to obtain a full baseline separation for the hundreds (or

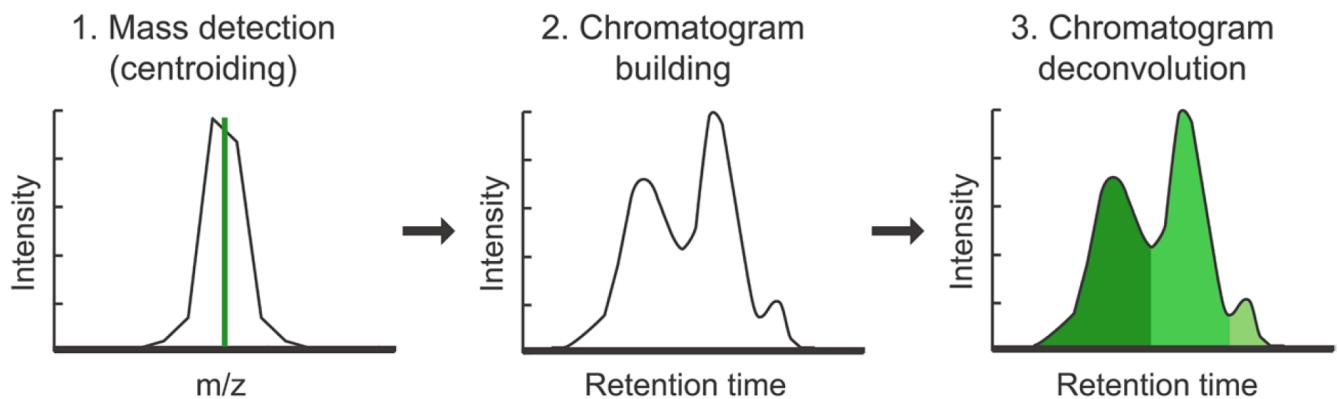
thousands) of analytes eluted through the column. The split of partially-overlapping and shoulder peaks into individual features is generally referred to as *chromatographic resolving* and is one of the most crucial steps of data processing. TO FINISH.

## 7.2 MZmine-specific terminology

### Masses and Features

In MS data processing, the term *mass* is normally used to refer to an individual signal in a mass spectrum, which corresponds to an ion detected by the mass spectrometer (see [Mass detection](#)).

In LC-MS, a *feature* is defined as a bounded, two-dimensional ( $m/z$  and RT dimensions) signal characterized by a pair of  $m/z$  and RT values and associated with the detected signal intensity. In the case of LC-IM-MS data, a feature is also characterized by the ion mobility value recorded for the ion (see [LC-MS and LC-IMS-MS data comparison](#)). MZmine 3 provides a selection of different algorithms for LC-(IM)-MS feature detection, depending on the nature of the MS data (e.g. mass accuracy and resolution). All the algorithms follow the same logic: EICs are constructed starting from each  $m/z$  value in the mass lists and subsequently deconvoluted into individual features (see figure). Further, additional information, such as isotope pattern, adduct type, etc. can be associated to the individual features.



### Mass list

In MZmine we call *mass list* the output of the [mass detection](#) module. A *mass list* is a list of  $m/z$  values (and corresponding signal intensities) found in each mass spectrum (MS or MSn) of each processed raw data file. Every mass spectrum contained in the raw file is processed individually and the signals exceeding the set noise threshold are included in the mass list. See [Mass detection](#) module.

### Feature list

In MZmine, *feature lists* are the output of the feature detection process (see [Masses and features](#)). The set of detected features in each LC-MS run is stored as a list, hence the name "feature list" (see, for example, [ADAP chromatogram builder](#) and [Local minimum resolver](#) for more details). Multiple feature lists can undergo further processing (e.g. feature alignment) which results in a table (often referred to as *feature table*) where samples are arranged in columns, features in rows and each entry contains the signal intensity detected for the corresponding feature, in the corresponding sample.

### Intra and inter-scan tolerances

To-do

#### 7.2.1 References

- Pluskal, T., Castillo, S., Villar-Briones, A. & Oresic, M. MZmine 2: Modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC Bioinformatics* (2010). DOI: 10.1186/1471-2105-11-395

- Pluskal, T. et al. Processing Metabolomics and Proteomics Data with Open Software: A Practical Guide, Chapter 7: Metabolomics Data Analysis Using MZmine (2020). DOI: 10.1039/9781788019880-00232
  - Smoluch M., Piechura K. Mass Spectrometry: An Applied Approach, Chapter 3: Basic Definitions (2019). DOI: 10.1002/9781119377368.ch3
- 

Last update: May 2, 2022 19:11:52

## 7.3 Ion mobility spectrometry terminology

---

### 7.3.1 Background

Ion-mobility mass-spectrometry, here simply referred to as ion-mobility (IM), is an analytical technique where ions are separated through a gas-filled mobility cell prior to the MS acquisition. Ions drift through the IM cell with different velocity based on their interaction with the buffer gas, which allows for the separation of different shaped molecules. Modern devices are able to perform IM separation on a millisecond timescale, typically within 10 to 100 ms. Thus, IM nicely fits in-between LC separation (~seconds timescale) and MS detection of TOF instruments (~microseconds timescale). This allows LC-IM-MS instruments to acquire several MS spectra during each [accumulation](#), without incurring sensitivity loss. For example, assuming a typical 100  $\mu$ s MS-acquisition time of TOF analyzers, around 1000 spectra can be recorded within 100 ms of IM separation. Therefore, as opposed to LC-MS, multiple MS (or MS<sub>2</sub>) spectra are associated to each RT in LC-IM-MS data. A more detailed explanation of LC-MS and LC-IMS-MS raw data structure is provided [here](#).

#### Trapped ion mobility spectrometry (TIMS)

TO-DO Trapped ion mobility spectrometry (TIMS) reverses the concept of traditional drift tube IM. Rather than moving ions through a stationary gas, TIMS holds ions stationary against a moving gas and then releases them according to their mobility. Video: <https://www.youtube.com/watch?v=cWjz32wky2A>

#### Time-dispersive ion mobility spectrometry (DTIMS and TWIMS)

TO-DO Time-dispersive IM devices include "traditional" drift tube (DTIMS) and travelling-wave (TWIMS) devices. In classic DTIM, ions migrate through an inert buffer gas under the influence of a weak electric field, whereas collisions with buffer gas molecules retard the progress of the ions. As larger ions have more collisions with the gas, they are more strongly retarded than their smaller counterparts. Thus, smaller ions, having a smaller cross section, arrive earlier at the detector than ions with a larger collisional cross section (CCS). The ion mobility K is then defined as the ratio of the analyte's steady-state net drift velocity to the applied electric field, and it is convention to calculate the reduced ion mobility K<sub>0</sub> at standard pressure and standard temperature, often reported as the inverse reduced ion mobility 1/K<sub>0</sub>.

### 7.3.2 Terminology

---

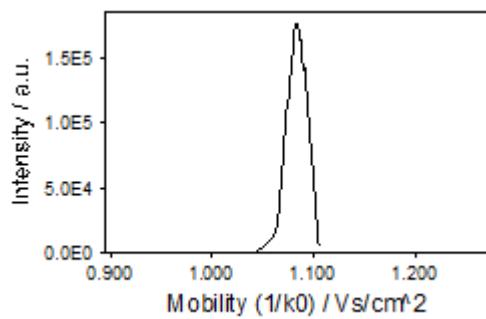
#### Accumulations, Mobility Scans and Frames

Although mainly used for TIMS, the term "*accumulation*" refers to the pack of ions gathered at the head of the IM device prior to the release and separation in the IM cell. As explained [above](#), since the accumulation-separation cycle typically last ~100 ms, multiple MS spectra (referred to as "*mobility scans*" in MZmine) are acquired during each cycle. The set of *mobility scans* collected during each IM separation constitutes a "*frame*". A *frame* can be seen as the IM separation of a single *accumulation*, along which multiple MS spectra are collected. Several *frames* are contained within one LC peak. Thus, the frame number are a natural unit to measure chromatographic RT. See [here](#) for more details.

--FIGURE HERE--

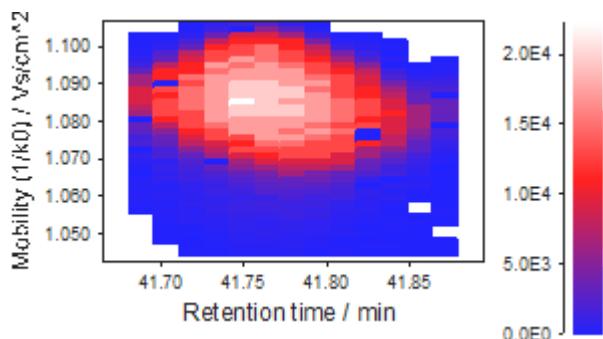
#### Mobilograms

A "*mobilogram*" represents the intensity of an m/z or m/z range along the mobility axis. A *mobilogram* may be build from multiple frames and summed or built from a single frame.



#### Ion mobility trace

An "ion mobility trace" basically represents a mobility resolved extracted ion chromatogram (EIC).



#### Cross Collisional Section

IM-derived CCS values can be used as an additional molecular descriptor to support the compound unknown identification process.(Paglia et al. 2014) However, the number of acquired spectra per run increases from several thousand to several million, requiring memory-efficient software and new processing algorithms.

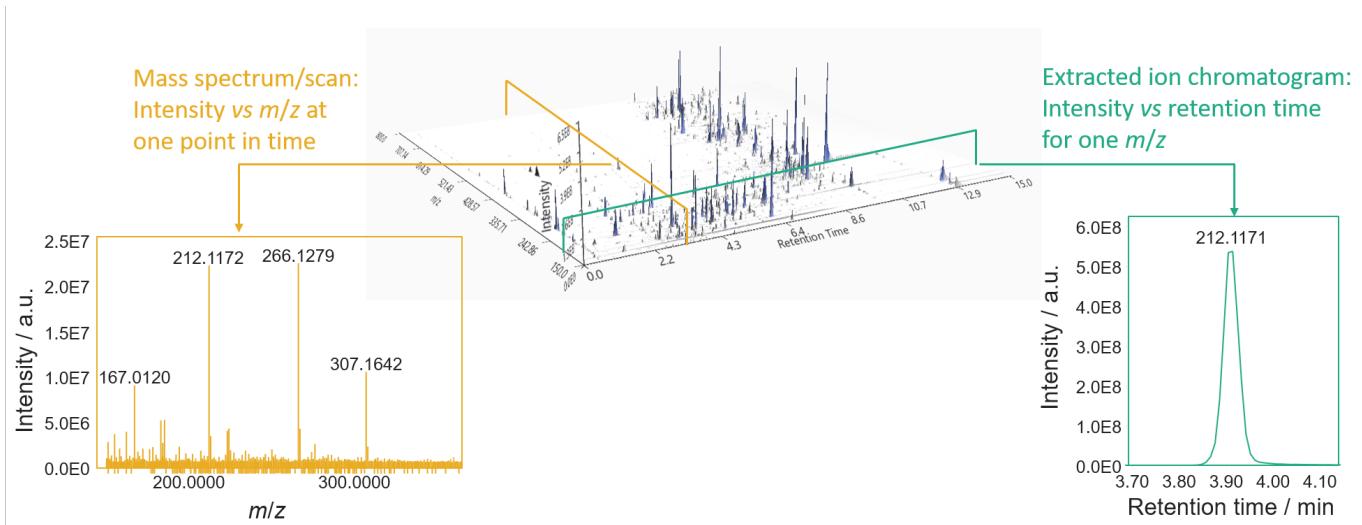
#### 7.3.3 References

- Meier, F., Brunner, A.D., Koch, S., Cox, J., Räther, O., Mann, M. Online Parallel Accumulation-Serial Fragmentation (PASEF) with a Novel Trapped Ion Mobility Mass Spectrometer. *Molecular & Cellular Proteomics* (2018). DOI: 10.1074/mcp.TIR118.000900

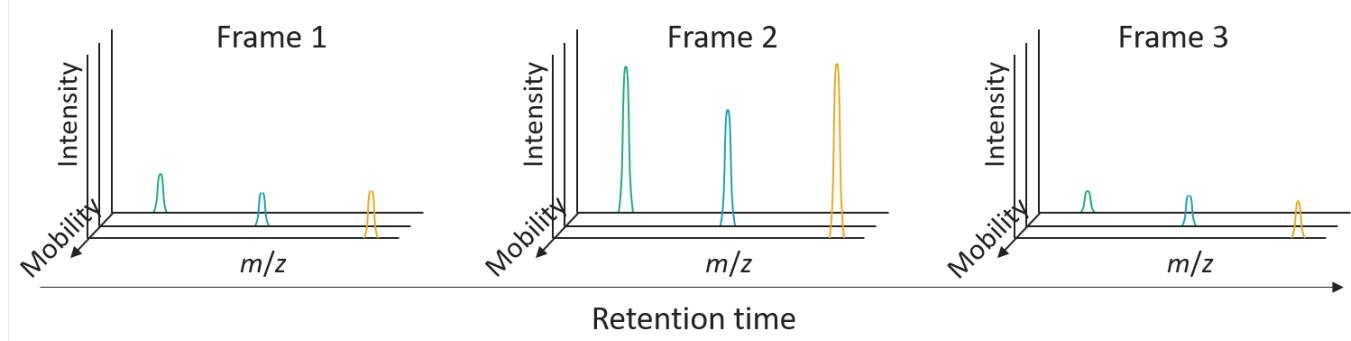
Last update: April 16, 2022 15:42:27

## 7.4 Graphical comparison of LC-MS and LC-IMS-MS data

Classic LC-MS data consists of three dimensions: m/z, intensity, and retention time. At every retention time, a whole mass spectrum is acquired (yellow). Putting all scans together creates a three-dimensional plane. By slicing the three-dimensional data at a single m/z (+- a tolerance), EICs can be created (green).



On the other hand, ion mobility resolved data consists of a three-dimensional data plane at each retention time. The three dimensions being m/z, intensity, and mobility (as drift time (ms) or inverse reduced mobility  $1/k_0$  [Vs/(cm<sup>2</sup>)]). The 3D projection of regular LC-MS data can be created by summing all mobility scans of a frame to create a frame spectrum. (see [Mobility scan merging](#))



Last update: March 10, 2022 15:42:38

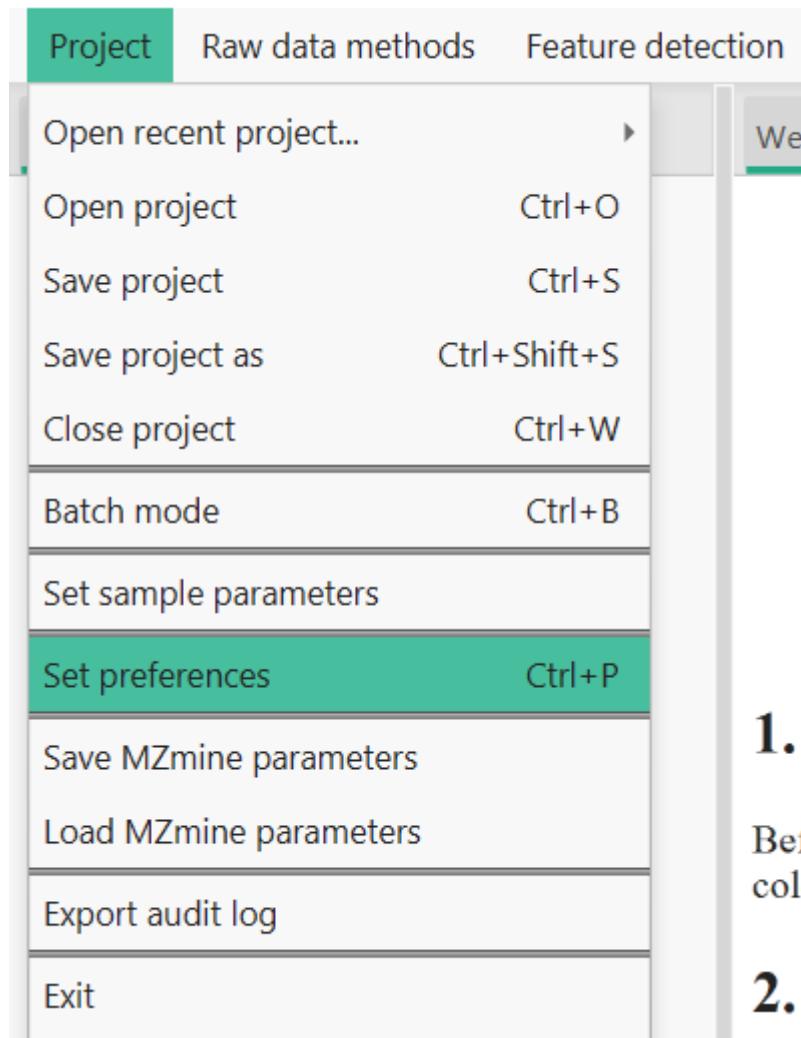
## 8. Performance options

---

This section contains information on how to tune MZmine 3 for different systems.

### 8.1 Preferences

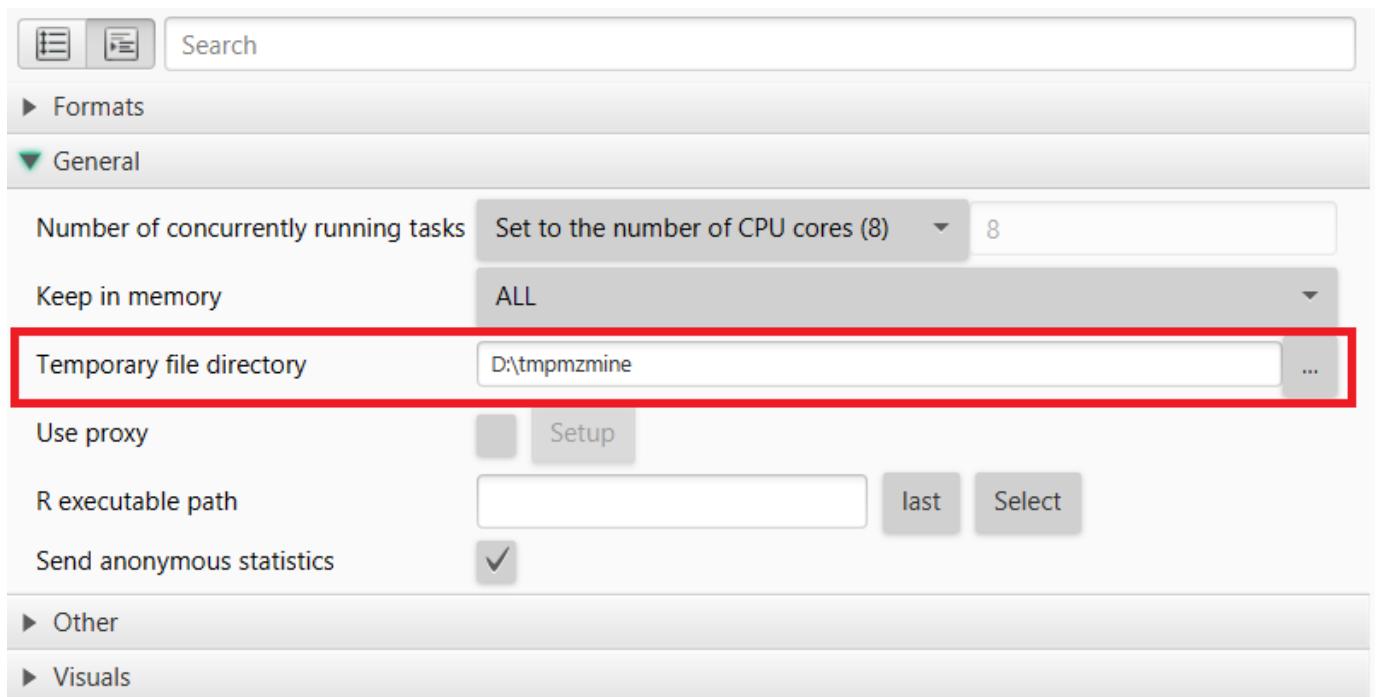
The preferences can be changed in MZmine's graphical user interface by accessing *File/Set preferences* from the menu. The choices will be stored in a (hidden) *.mzmine3.conf* file in the user's home directory (Windows: *C:\Users\USERNAME*) once MZmine is closed.



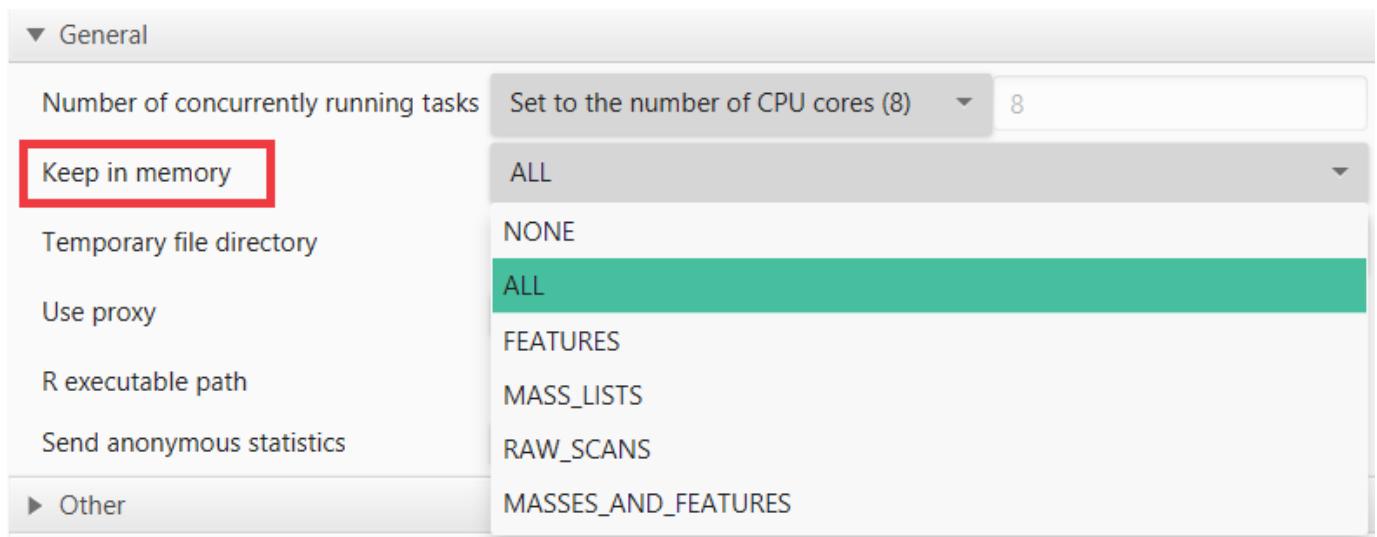
#### 8.1.1 Temporary files

MZmine will create multiple temporary files at various times of the processing stage, e.g., when importing spectral data, running mass detection, or creating feature lists. These files will be stored in a folder that can be specified in the preferences.

We recommend putting this folder on an SSD drive, ideally an M.2 for the best performance. The temporary files will be deleted when MZmine is closed (Mac & Linux) or when a new session is started (Windows).



### 8.1.2 Memory options



The parameter **Keep in Memory** defines what data is kept in memory (RAM) or otherwise memory mapped to the temp directory.

- Generally this setting should be *none* (**default**).
- If memory is no issue this option might be set to *all* process all spectral and feature data in memory.
- The option *masses\_features* keeps centroid mass lists and features in memory while memory mapping raw spectral data.
- The option *mass\_lists* will keep only mass lists in RAM, while memory mapping the raw spectral data and features.

## 8.2 Logs

Currently, the logs are written to an *mzmine\_0\_0.log* file in the user's home directory. Please submit your log files together with any issues on [GitHub](#).

Last update: April 8, 2022 08:12:41

## 9. Command-line arguments

---

Command-line arguments offer a variety of options that generally override the corresponding parameters in the preferences.

### Windows

An easy way to start MZmine with arguments is to create a shortcut to the MZmine.exe, right-click, and define the target with additional arguments. This example runs MZMine in batch mode (headless), imports the specified batch file, overrides the memory management to **none** (which is the default), effectively using memory mapping to store and access spectral, centroid, and feature data from temporary files stored in the defined temp directory. By leaving out the *memory* or *temp* arguments, the values stored in the current *preferences* file will be used, or the default values if no *preferences* file was found.

#### Start MZmine batch with memory mapping (DEFAULT)

```
"C:\Program Files\MZmine\MZmine.exe" -batch "D:\batch\my_batch_file.xml" -memory none -temp "D:\tmpmzmine"
```

#### Start MZmine batch on machines with enough memory (RAM) with -memory all

```
"C:\Program Files\MZmine\MZmine.exe" -batch "D:\batch\my_batch_file.xml" -memory all -temp "D:\tmpmzmine"
```

### 9.0.1 Argument table

Argument	Options (default)	Description
-batch	a path, e.g. "D:\batch.xml"	Path to batch file
-memory	<b>none</b> , all, features, centroids, raw, masses_features	Defines what data is kept in memory (RAM) or otherwise memory mapped to the temp directory. Generally this setting should be <i>none</i> . If memory is no issue this option might be set to <i>all</i> process all spectral and feature data in memory. The option <i>masses_features</i> keeps centroid mass lists and features in memory while memory mapping raw spectral data.
-temp	a path, e.g., "-temp "D:\tmpmzmine\"	The defined directory should be on a fast drive (usually SSD > HDD > network drive) with enough free space. Local drives are usually preferred. MZmine uses memory mapping to efficiently store and access spectral and feature data. This can lead to a considerable temporary consumption of disk space. Make sure that the selected drive has enough space (maybe 20 GB + 1 GB/10 files; generously over estimated).

Last update: April 8, 2022 08:12:41

# 10. Contribute

---

## 10.1 How to contribute

### 10.1.1 Contribute to the MZmine documentation

#### 1. Make a GitHub Account

You'll need to make a [GitHub Account](#).

#### 2. Click Edit Button on Page You Want to Edit

##### MZmine 3 Documentation

[Home page](#)  
[Main window overview](#)  
[LC-MS workflow](#)  
[LC-IMS-MS workflow](#)  
[Raw data visualisation](#)

##### LC-IMS-MS Workflow

##### Supported formats

- Vendor formats: \*

  - .tdf (Native Bruker LC-IMS-MS and MALDI-IMS-MSI format) \*
  - .tsf (Native Bruker MALDI-IMS-MS (single shot) format)

- .mzML \*

  - Created via MSConvert from native Bruker data \*
  - Created via MSConvert from native Waters data

Table
Suoppc
Edit this page
Backg
termir
Mot
form
Fran
Mot
Ion i
Raw d
Raw
Mas
Se

#### 3. Fork the Repository When Prompted (only the first time)



You need to fork this repository to propose changes.

Sorry, you're not able to edit this repository directly—you need to fork it and propose your changes from there instead.

[Fork this repository](#)

[Learn more about forks](#)

#### 4. Make the Edits in MarkDown

mzmine\_documentation / docs / Ion-mobility-data-proc Cancel changes

Spaces 3 Soft wrap

```

1 # LC-IMS-MS Workflow
2 ## Supported formats
3
4 * Vendor formats:
5 *
6     * .tdf (Native Bruker LC-IMS-MS and MALDI-IMS-MSI format)
7 *
8     * .tsf (Native Bruker MALDI-IMS-MS (single shot) format)
9 * .mzML
10 *
11     * Created via [MSConvert](https://proteowizard.sourceforge.io/download.html) from native Bruker
12     data
13 *
14     * Created via [MSConvert](https://proteowizard.sourceforge.io/download.html) from native Waters
15     data
16
17 **Note**: mzML via MSConvert from Agilent raw data can be imported, but certain restrictions might
18 hinder processing workflows due to the nature of the raw data format.
19
20
21 ***
22
23 ## Background information and terminology
24
25 Since ion mobility spectrometry (IMS) resolved data is more complex due to the additional dimension
26 when compared to regular LC-MS data, some terms shall be clarified before going into details of the
27 processing steps.
28
29 ### Mobility separation and data format
30
31 Ion mobility separation usually occurs on the millisecond timescale, fitting nicely in-between
32 liquid chromatography (LC) (few seconds per chromatographic peak) and mass spectra acquisition of
33 TOF instruments (several micro seconds). Therefore, the mobility dimension can be resolved by
34 acquiring multiple spectra during a mobility separation (e.g. 1000 spectra per 100 ms). This leads
35 to multiple mass spectra acquired at one IMS accumulation. Thus, at one retention time, multiple
36 spectra are acquired. A detailed comparison of LC-MS and LC-IMS-MS raw data can be

```

Attach files by dragging & dropping, selecting or pasting them.

## 5. Propose Changes

Please describe the change you are making.

### Commit changes

update mobility resolving step

add msms pairing description in mobility resolving step

steffen.heuckeroth@gmx.de

Choose which email address to associate with this commit

- o Commit directly to the `master` branch.
- Create a new branch for this commit and start a pull request. Learn more about pull requests.

## 6. Create Pull Request

The screenshot shows the GitHub repository page for 'mzmine / mzmine\_documentation'. The 'Code' tab is selected. A yellow banner at the top right indicates 'SteffenHeu-patch-1 had recent pushes 1 minute ago' and contains a green button labeled 'Compare & pull request'.

## 7. Finalize Pull Request with Description

The screenshot shows the pull request description editor. The base branch is set to 'master'. The merge status is shown as 'Able to merge. These branches can be automatically merged'. The title of the pull request is 'update mobility resolving step'. The description text is 'add msms pairing description in mobility resolving step'. There is a note at the bottom left: 'Attach files by dragging & dropping, selecting or pasting them.' A green 'Create pull request' button is at the bottom right.

## 10.1.2 Creating a new page

**Follow steps 1 - 3.**

**Navigate to mzmine\_documentation/docs in your fork and create a new file**

The screenshot shows a GitHub repository interface for the 'mzmine\_documentation' branch. The top navigation bar includes 'master', 'mzmine\_documentation / docs /', 'Go to file', 'Add file', and a three-dot menu. A blue button labeled 'Create new file' is highlighted. Below the navigation is a list of contributions by user 'SteffenHeu'. The contributions are:

File/Action	Description	Time Ago
try png logo	try png logo	20 hours ago
add all documents to nav	add all documents to nav	yesterday
Update Ion-mobility-data-processing-workflow.md	Update Ion-mobility-data-processing-workflow.md	22 hours ago
Add credit	Add credit	3 months ago
typos, fix links	typos, fix links	2 days ago
fix remaining dead links	fix remaining dead links	yesterday
fixed layout	fixed layout	last month
add page contributors, add gnps acknowledgements	add page contributors, add gnps acknowledgements	3 months ago

#### Follow steps 4 - 7.

##### 10.1.3 Page Contributors

[SteffenHeu](#)

This page was adapted from the [GNPS documentation](#).

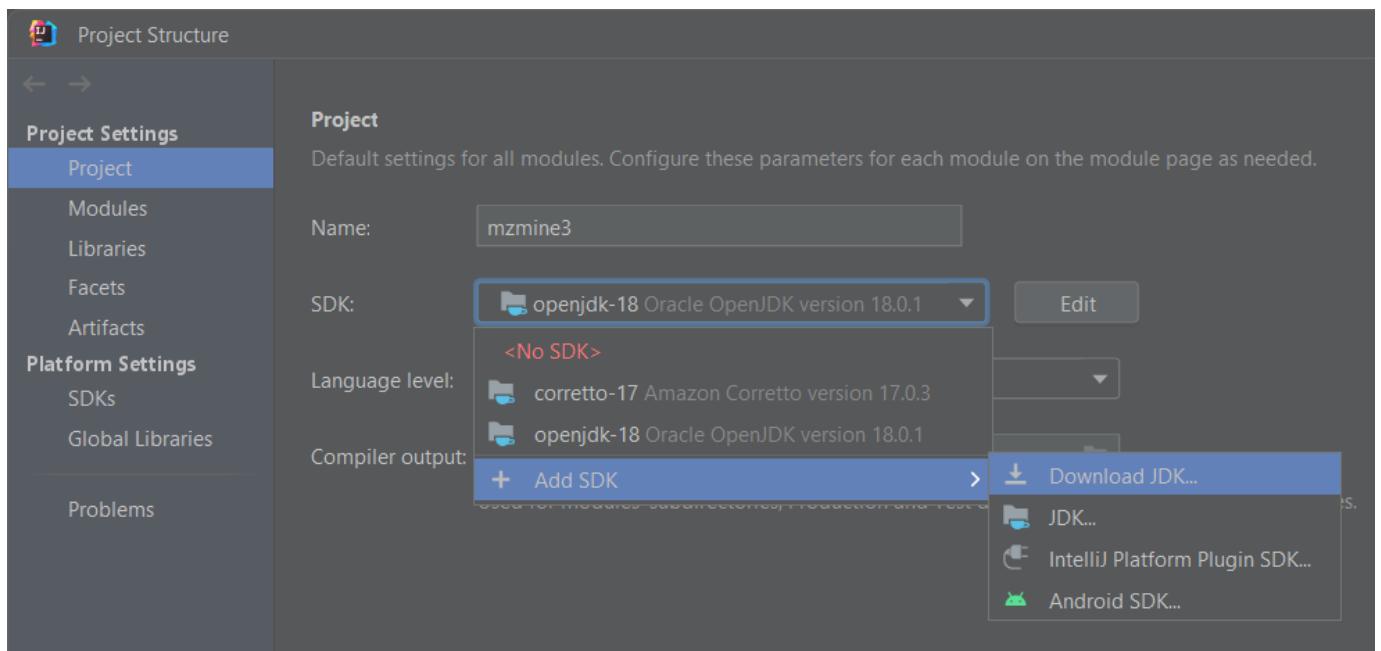
---

Last update: April 5, 2022 13:22:07

## 10.2 Development in IntelliJ

### 10.2.1 Set up

1. Fork the mzmine (<https://github.com/mzmine/mzmine3>) GitHub repository (needs free GitHub account) (See <https://help.github.com/en/github/getting-started-with-github/fork-a-repo>)
2. Download & Install IntelliJ IDEA (there is a free educational license for students and teachers) from <https://www.jetbrains.com/idea/download/>
3. Download & Install the current JDK. We recommend the OpenJDK. However, you can also use any other distribution, e.g., the Oracle JDK. This can be done from within IntelliJ. Open *File/Project Structure* (CTRL+ALT+SHIFT+S) and select SDKs and add the latest JDK with the +button:



4. Add your GitHub account via **Settings/Version Control/GitHub** +button. Below exemplified with the Log in with Token... option: - Log in with Token... **Generate** - redirects to GitHub - Make sure to select the **Workflow** scope to avoid conflicts that arise from changing GitHub actions

**Settings / Developer settings**

## New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

**Note**

IntelliJ IDEA GitHub integration plugin

What's this token for?

**Expiration \***

30 days      The token will expire on Fri, Jun 10 2022

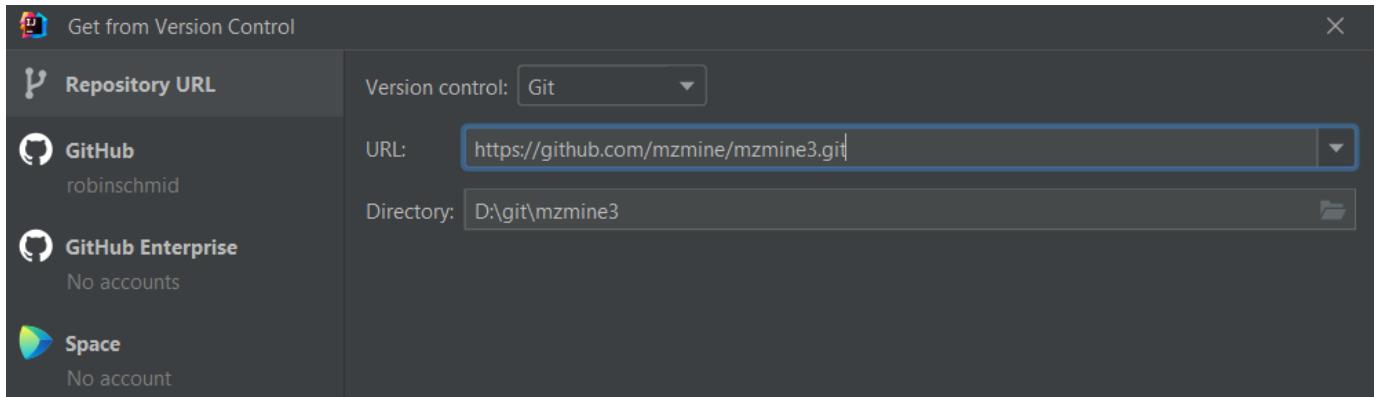
**Select scopes**

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

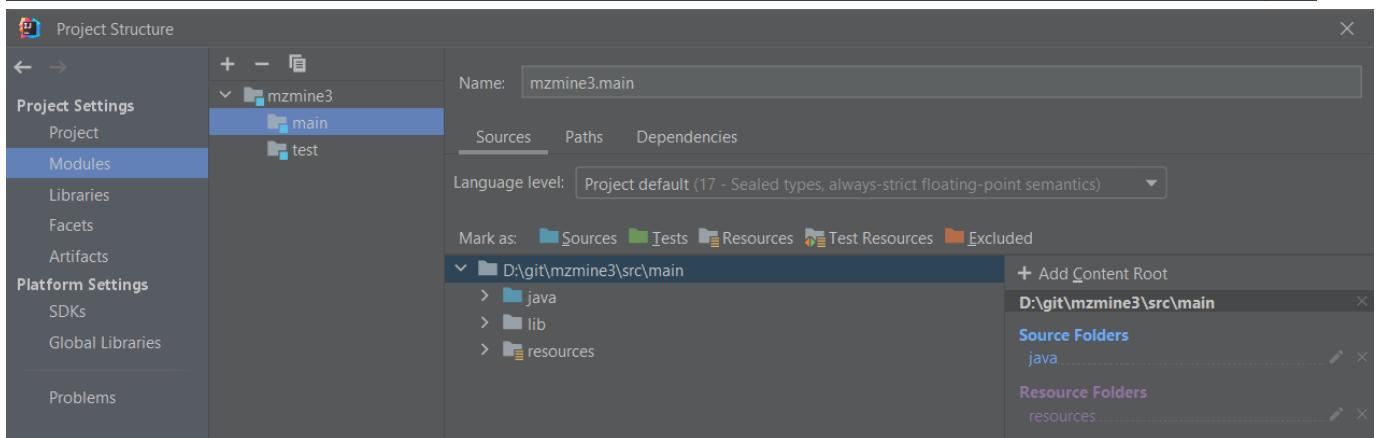
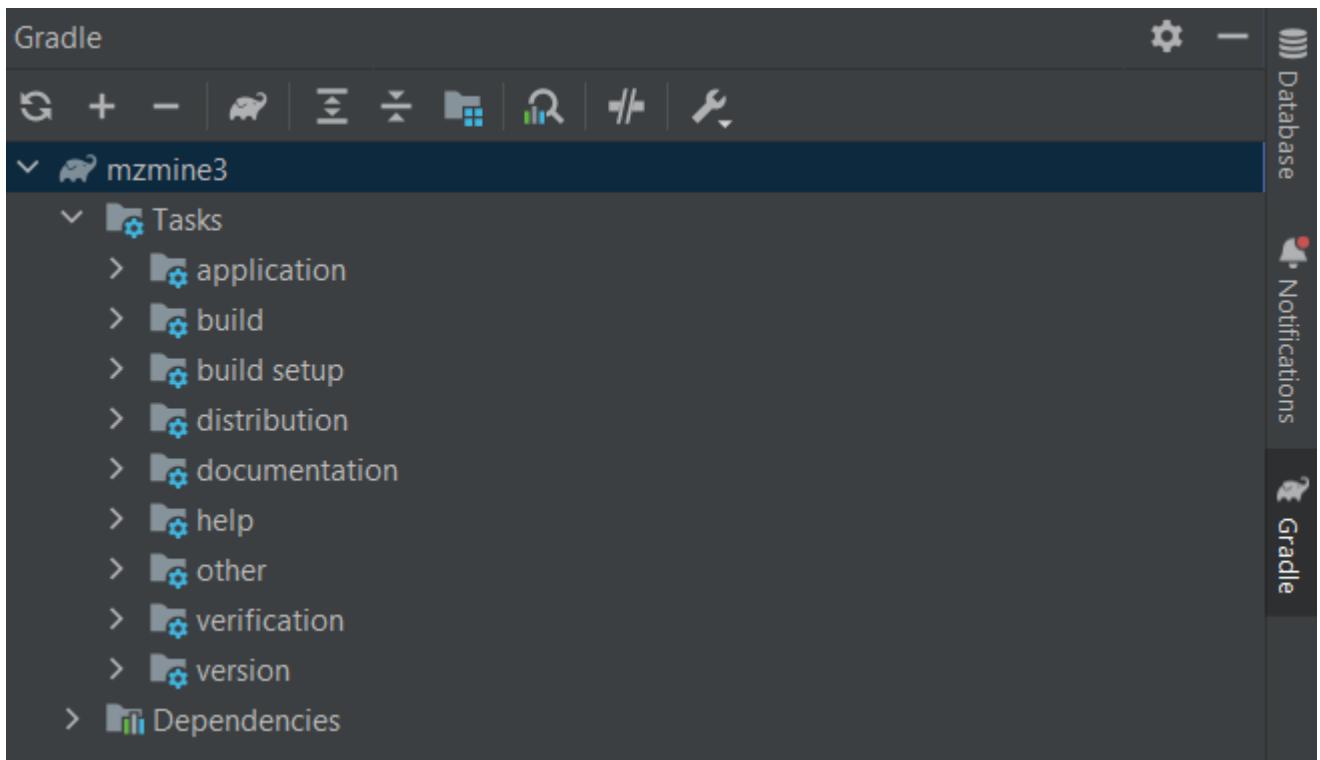
<input checked="" type="checkbox"/> <b>repo</b>	Full control of private repositories
<input checked="" type="checkbox"/> <b>repository_status</b>	Access commit status
<input checked="" type="checkbox"/> <b>repo_deployment</b>	Access deployment status
<input checked="" type="checkbox"/> <b>public_repo</b>	Access public repositories
<input checked="" type="checkbox"/> <b>repo_invite</b>	Access repository invitations
<input checked="" type="checkbox"/> <b>security_events</b>	Read and write security events
<input checked="" type="checkbox"/> <b>workflow</b>	Update GitHub Action workflows
<input type="checkbox"/> <b>write_packages</b>	Upload packages to GitHub Package Registry
<input type="checkbox"/> <b>read_packages</b>	Download packages from GitHub Package Registry

5. Clone GitHub project via version control: **File/New/Project from version control** use your user name to get your fork:

[https://github.com/YOUR USERNAME/mzmine3.git](https://github.com/YOUR_USERNAME/mzmine3.git)



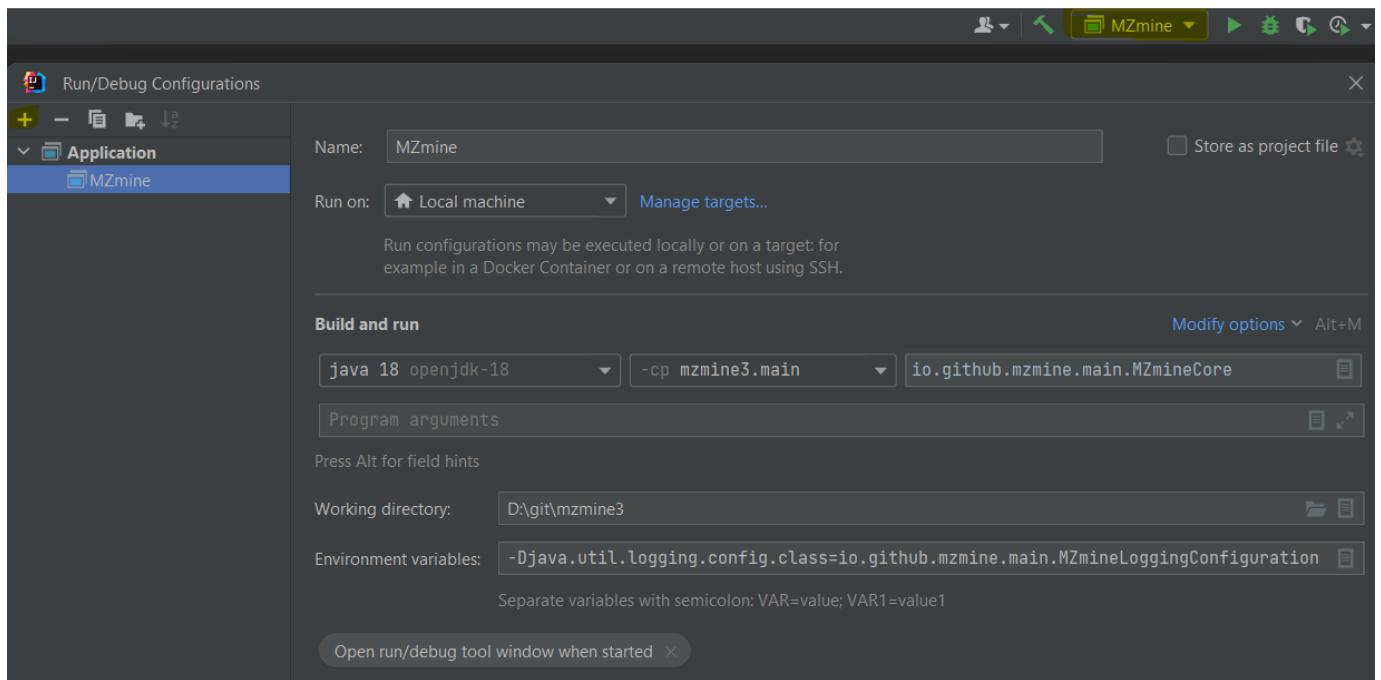
1. Make sure that gradle runs and updates the project. Otherwise, click on **Reload Gradle Project**. Now the project structure (CTRL+SHIFT+ALT+S) should show the source, test, and resource folders which are described in the build.gradle.



1. Click on Add a Configuration. Select “Application” from the template list). via the + button (don’t just edit the template):

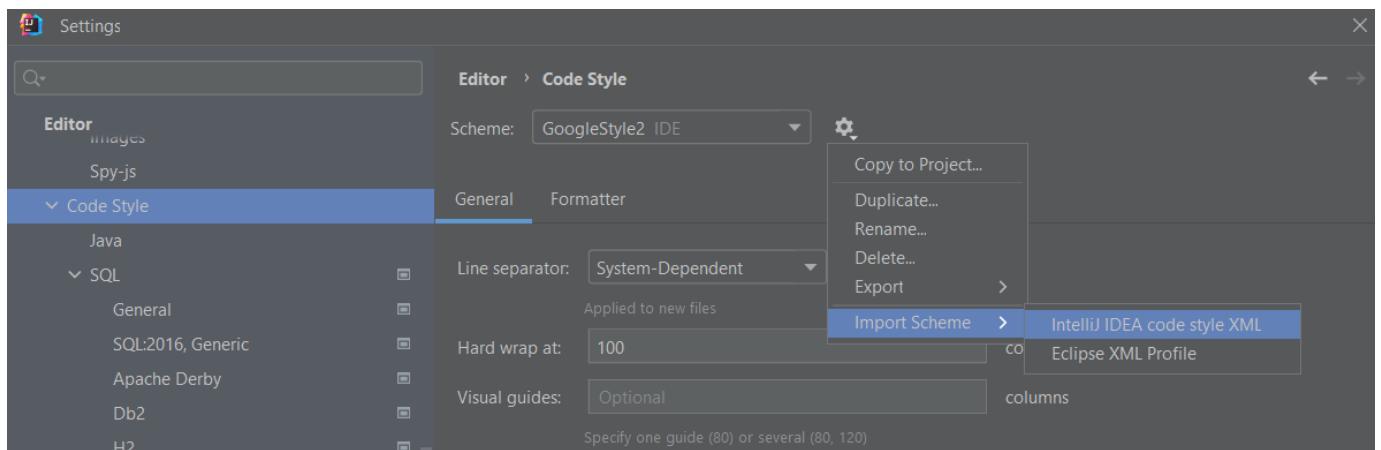
- Main class: `io.github.mzmine.main.MZmineCore`
- Environment var: `-Djava.util.logging.config.class=io.github.mzmine.main.MZmineLoggingConfiguration -Xmx12G`

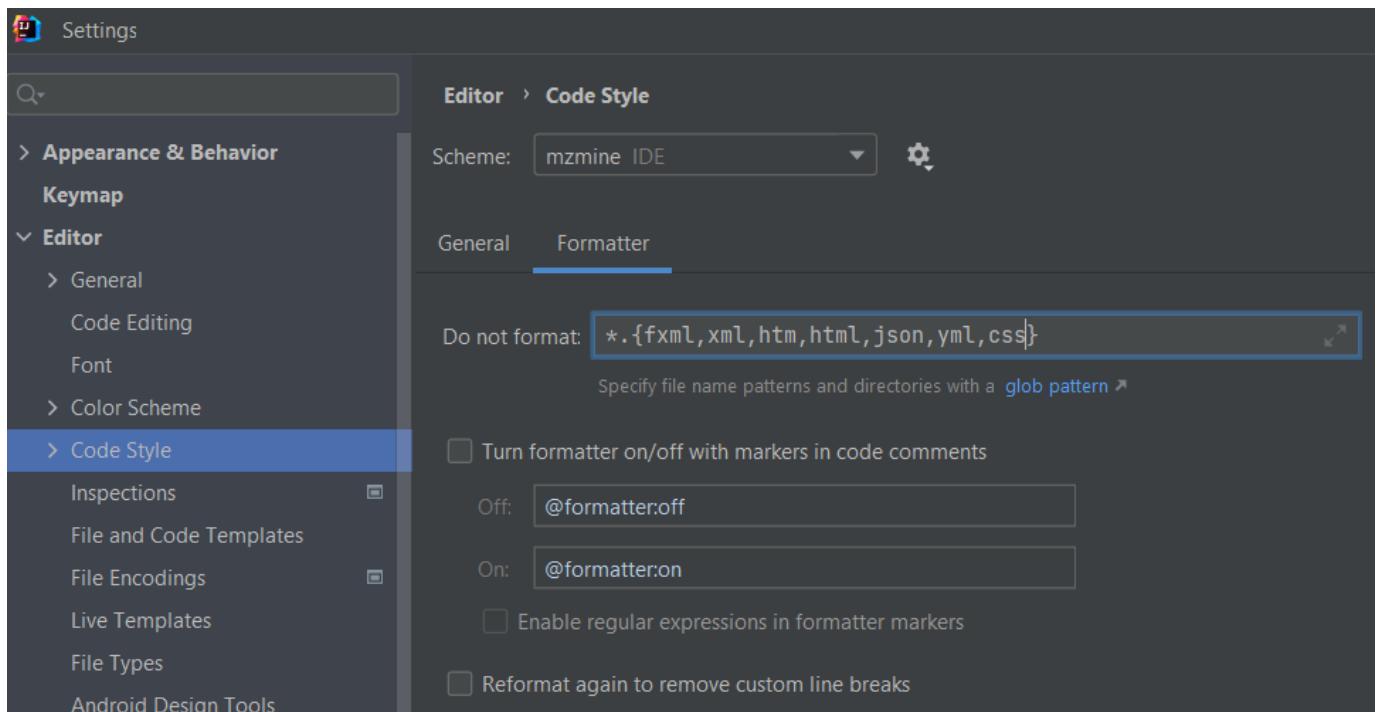
2. Run or debug with this configuration



## 10.2.2 Code formatter

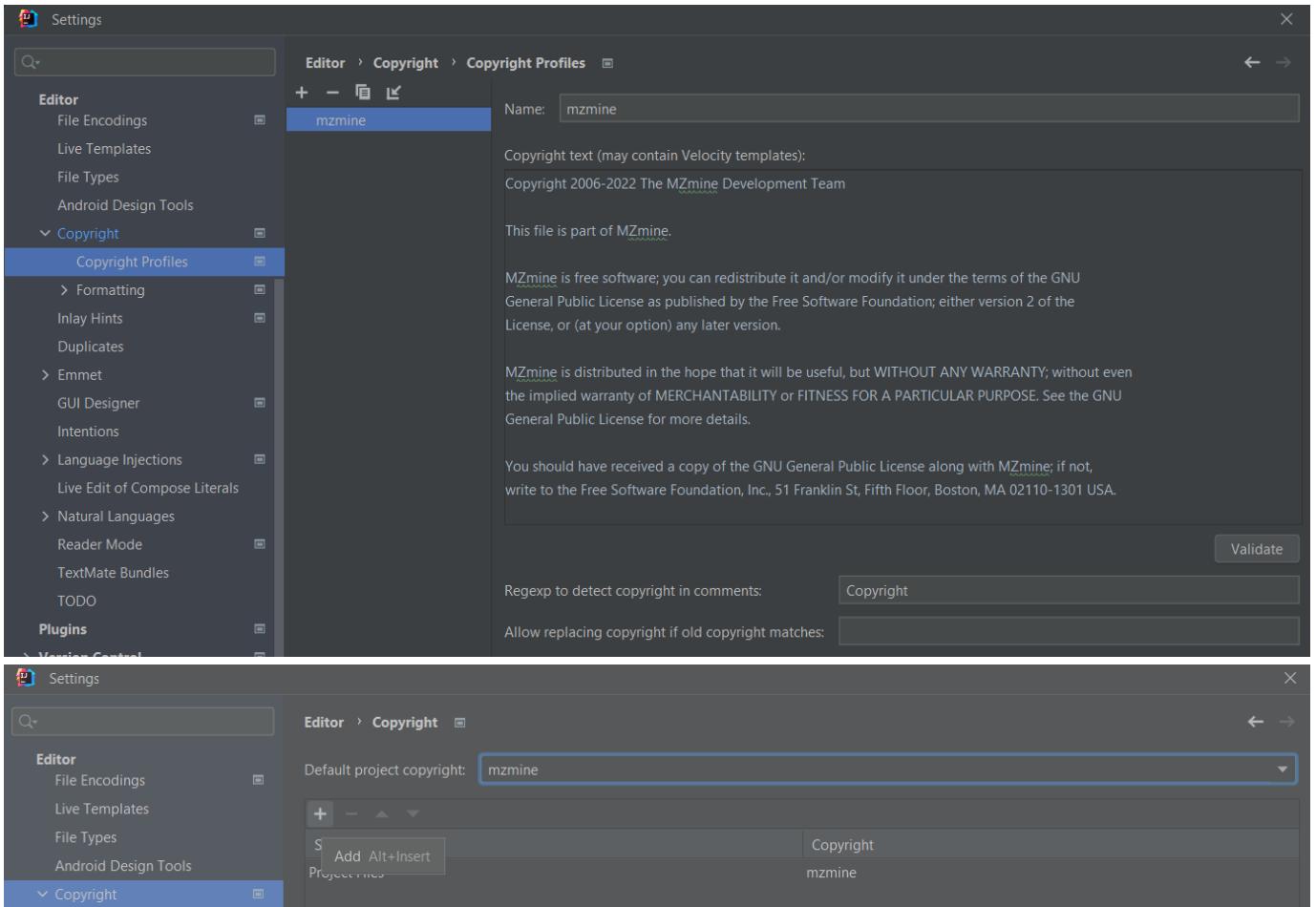
The code formatter is important for contributions to the main version of MZmine and each java file should contain the license header. 1. Import the style format from `mzmine-intellij-code-formater.xml` 2. Apply format to any file with **CTRL+ALT+L** 3. Can also be applied on each file saved or each committed change (see below) 4. Better exclude specific file formats from being formatted (see below screenshot 2)





### 10.2.3 Copyright header

1. Add the license header to each file - the easiest way is to add the copyright profile to intelliJ
2. Import the copyright from `mzmine_intellij_licence_header.xml` or create a new one with the exact text specified in `license_header.txt`
3. Add a new scope for all project files
4. Apply after file save or commit operation or run the **Update copyright...** action



## 10.2.4 Useful settings

### Faster building

1. Activate auto building (consumes more resources)
2. Set **Settings/Gradle/build and run** to IntelliJ
3. Activate HotSwap to automatically load changed classes during debugging

**Settings**

compiler

**Build, Execution, Deployment > Compiler**

Resource patterns: `*.java;!?*.form;!?*.class;!?*.groovy;!?*.scala;!?*.flex;!?*.kt;!?*.clj;!?*.aj`

Use ; to separate patterns and ! to negate a pattern. Accepted wildcards: ? — exactly one symbol; \* — zero or more symbols; / — path separator; /\*\* — any number of directories; <dir\_name>:<pattern> — restrict to source roots with the specified name

Clear output directory on rebuild

Add runtime assertions for notnull-annotated methods and parameters [Configure annotations...](#)

Automatically show first error in editor

Display notification on build completion

Build project automatically (only works while not running / debugging)

Compile independent modules in parallel (may require larger heap size)

Rebuild module on dependency change

Shared build process heap size (Mbytes):

Shared build process VM options:

User-local build process heap size (Mbytes) (overrides Shared size):

User-local build process VM options (overrides Shared options):

**Settings**

Build, Execution, Deployment > Build Tools > Gradle

General settings

Gradle user home:  [...](#)

Generate \*.iml files for modules imported from Gradle Enable if you have a mixed project with IntelliJ IDEA modules and Gradle modules so that it could be shared via VCS

Gradle projects

mzmine3  Download external annotations for dependencies

Build and run

By default IntelliJ IDEA uses Gradle to build the project and run the tasks.

In a pure Java/Kotlin project, building and running by means of the IDE might be faster, thanks to optimizations. Note, that the IDE doesn't support all Gradle plugins and the project might not be built correctly with some of them.

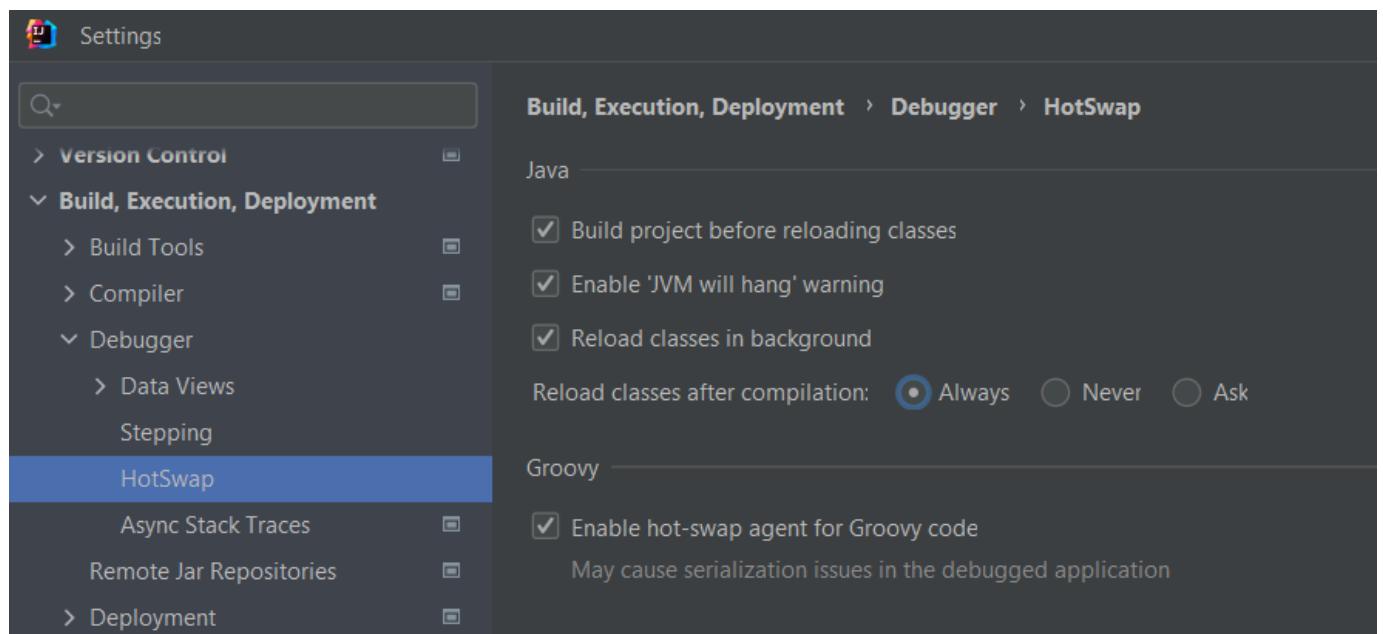
Build and run using:

Run tests using:

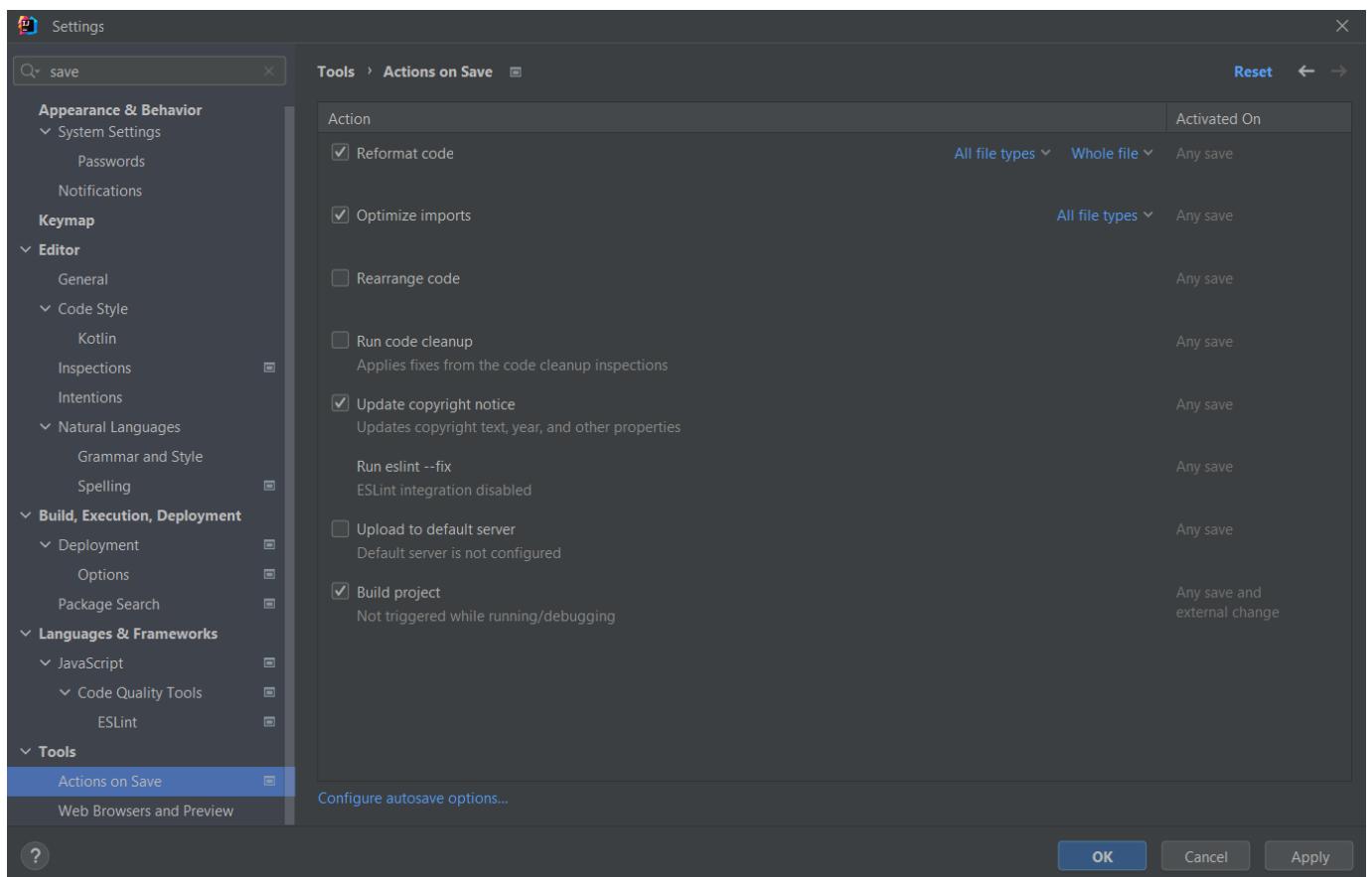
Gradle

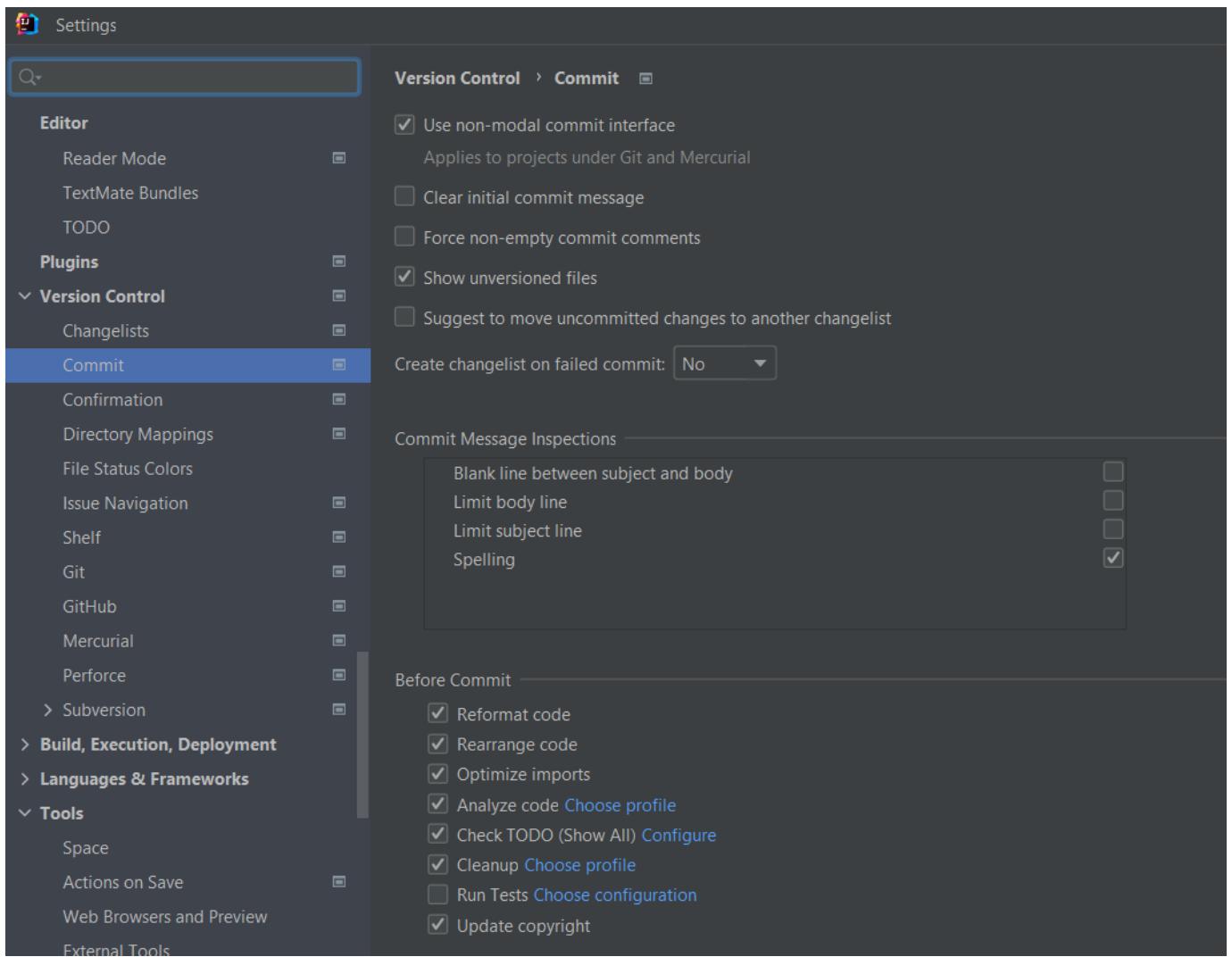
Use Gradle from:

Gradle JVM:

**Save and commit actions: Apply copyright, format, etc**

Select any actions to perform when a file is saved or committed.





## Live templates

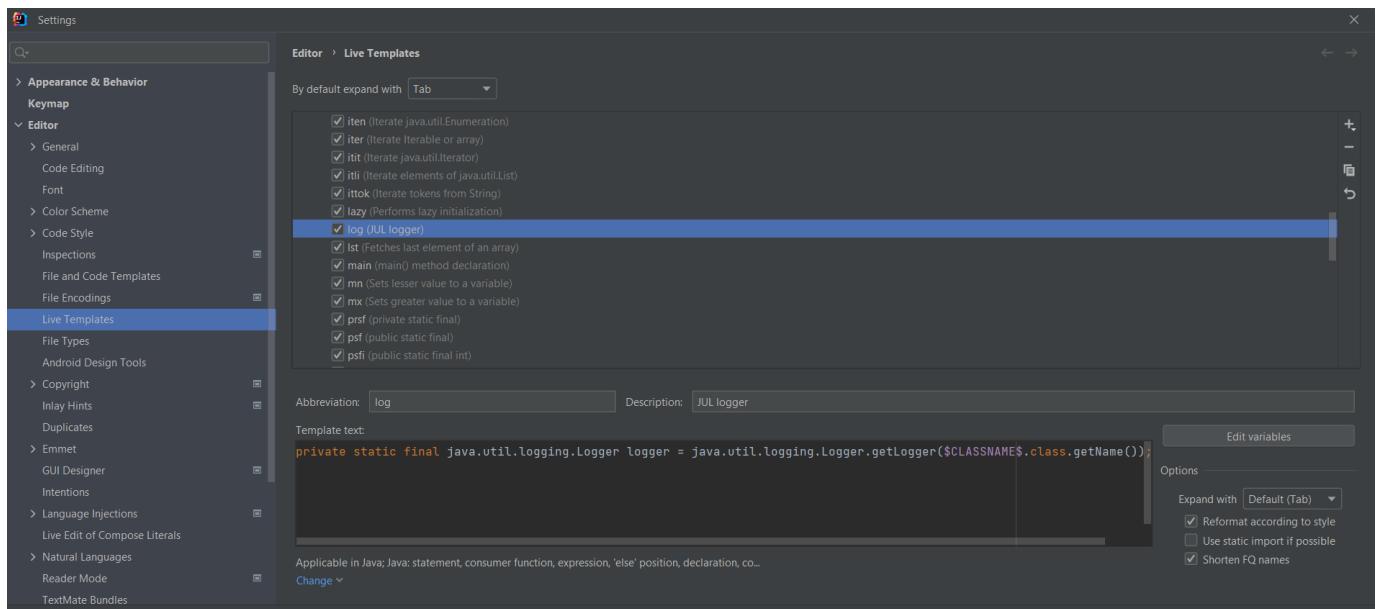
Live templates are used to add custom entries to the code completion, e.g., to quickly create a class-specific logger. Access **Settings/Editor/Live Templates** (CTRL+ALT+S on Windows) and add a new template (+ button). Define the abbreviation to trigger autocompletion at this statement, define the template text, and change the target context ("Java") at the bottom. The template below generates a logger after typing log and pressing CTRL+SPACE. The variable **\(CLASS\_NAME\)** was set to represent the current className() under **Edit variable**.

The template:

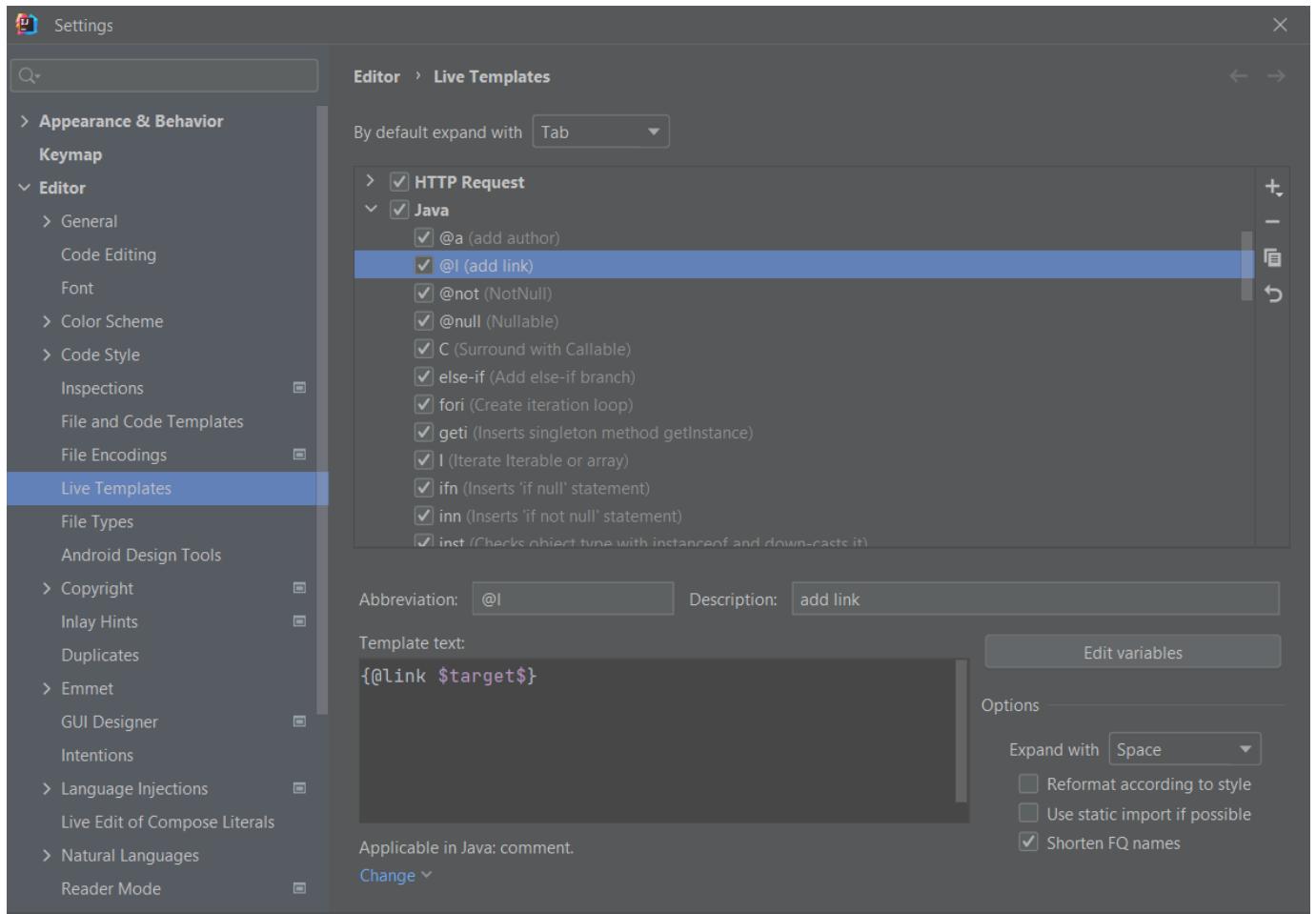
```
private static final java.util.logging.Logger logger = java.util.logging.Logger.getLogger($CLASSNAME$.class.getName());
```

Generates the output in class Scan:

```
private static final Logger logger = Logger.getLogger(Scan.class.getName());
```



Another example to create Javadoc links for @l . The variable (here \\$(target\)) places the cursor.



## 10.2.5 Troubleshooting

### Correct JDK selection

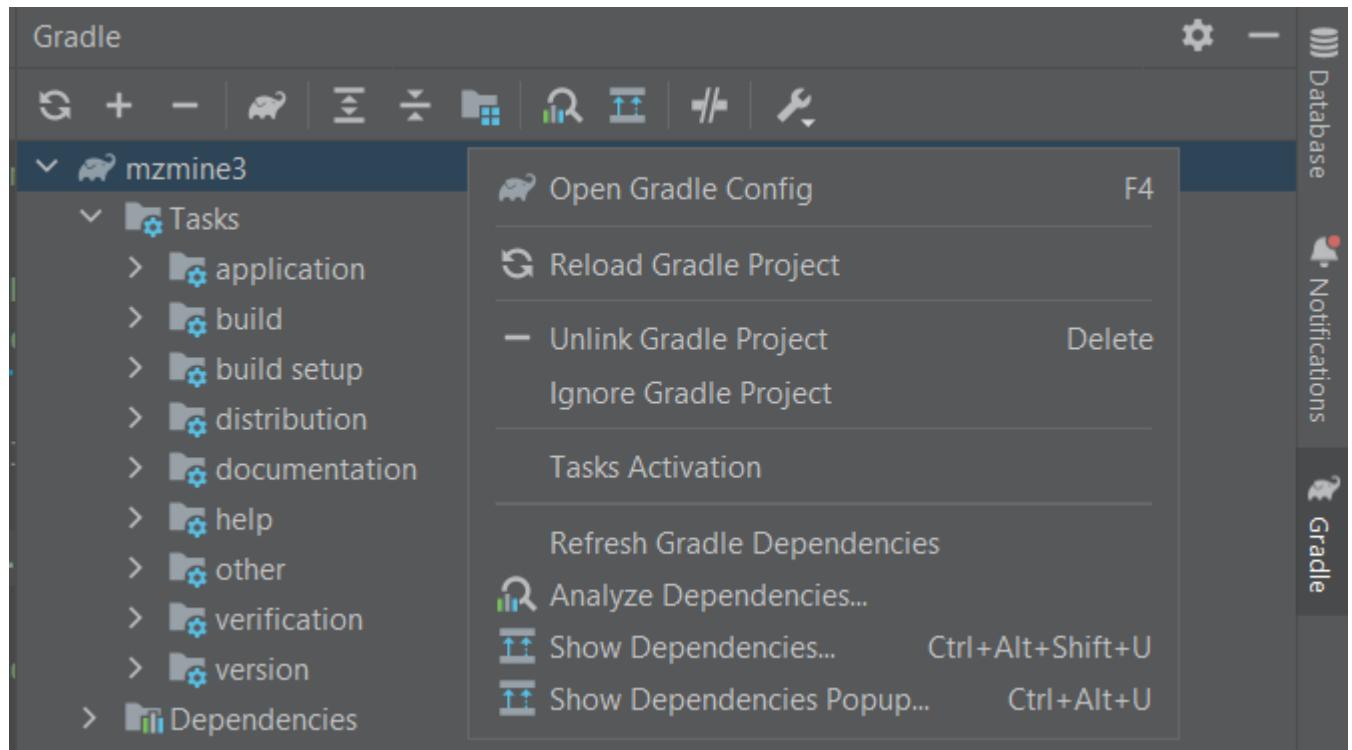
Make sure the correct JDK is set in these places: 1. File/Project Structure/SDKs 2. File/Settings/Build, Execution, Deployment/Build Tools → Gradle → Gradle JVM → “Project SDK” this will update automatically if the project SDK changes. 3. File/Settings/Build, Execution, Deployment/Java Compiler → Project Bytecode version → 17 (for JDK 17) Correct run configuration: Select Default JRE (this will update with the project sdk) or select the correct one manually.

### Could not target platform

Error: When building via gradlew: “Could not target platform: ‘Java SE 17’ using tool chain: ‘JDK 13 (13)’” Solution (Windows): Set the JAVA\_HOME environment variable to the JDK 17 root directory. See [https://docs.oracle.com/cd/E19182-01/821-0917/inst\\_jdk\\_javahome\\_t/index.html](https://docs.oracle.com/cd/E19182-01/821-0917/inst_jdk_javahome_t/index.html)

### Gradle project not imported

If gradle tool window is not shown: 1. To import the Gradle project navigate to the build.gradle in the project tool window right click → import gradle project. The gradle tool window should now be visible. 2. To update the imports click the update gradle project button in the gradle tool window



Last update: June 1, 2022 08:55:52

## 11. Acknowledgements

---

We would like to point out that this wiki was set up in tight collaboration with the [GNPS](#) staff. We highly appreciate your help!

### 11.1 Related projects

---

- [GNPS](#)
- [SIRIUS](#)

### 11.2 Libraries we use in MZmine

---

- [Apache XML Graphics](#) - EPS image export
- [Chemistry Development Kit](#) - Isotope pattern and molecular calculations
- [Freehep](#) - EMF image export
- [Google Guava](#) - Utility classes
- [JDK Documentation](#)
- [JChemPaint](#) - 2D molecule visualization
- [JFreeChart](#) - TIC, Spectra and 2D visualizers
- [J Mol](#) - 3D molecule visualization
- [jmzml](#) - mzML file import
- [jmzTab](#) - mzTab file import and export
- [NetCDF-Java](#) - NetCDF file import
- [VisAD](#) - 3D visualizer
- [WEKA](#) - Clustering and other machine learning algorithms
- [Bruker TDF SDK](#) - Native tdf/tdf file import (requires VC++ 2017 redist.)
- [Thermo raw file parser](#) - Native Thermo raw import

---

Last update: March 10, 2022 15:35:04