TOPP and TOPPView Tutorial

Version: 1.1

Contents

1 TOPP Tutorial

1.1 General introduction

This tutorial will give you a brief overview of the most important TOPP tools. First, we explain some basics that you will need for every TOPP tool, then we show several example pipelines.

1.1.1 File formats

The TOPP tools use the HUPO-PSI standard format mzData 1.05 as input format. In order to convert other open formats (DTA, mzXML, ANDI/MS) to mzData, a file converter is provided by TOPP.

Proprietary MS machine formats are not supported. If you need to convert these formats to mzData or mzXML, please have a look at the `SASHIMI project page` or contact your MS machine vendor.

mzData covers only the output of a mass spectrometry experiment. For further analysis of this data several other file formats are needed. The main file formats used by TOPP are:

- **mzData** The HUPO-PSI standard format for mass spectrometry data.
- **featureXML** The OpenMS format for quantitation results.
- **consensusXML** The OpenMS format for alignment of peak and feature data.
- **featurePairsXML** The OpenMS format for feature pairs.
- **idXML** The OpenMS format for protein and peptide identification.

Documented schemas of the OpenMS formats can be found at `http://open-ms.sourceforge.net/schemas/` .

For identification results in idXML format, there is an XSLT-script that you can use to visualize the identifications in a web browser. Direct your browser to `http://open-ms.sourceforge.net/schemas/IdXML.xsl` and store the XSLT-script in a directory of your choice, say, `C:\scripts\IdXML.xsl`. Add a line of the form

```
<?xml-stylesheet type="text/xsl" href="*PATH*IdXML.xsl"?>
```

after the first line of the idXML file you want to inspect. This can be done using a standard text editor. Replace ∗PATH∗ with the path to the directory which contains the XSLT script IdXML.xsl. For example, if the directory is C:\scripts\ just exchange ∗PATH∗ with

1.2 Example 1: File Handling

1.2.1 General information about peak and feature maps

If you want some general information about a peak or feature map, use the **FileInfo** tool.

- It can print RT, m/z and intensity ranges, the overall number of peaks, and the distribution of MS levels
- It can print a statistical summary of intensities
- It can print some meta information
- It can validate XML files against their schema
- It can check for corrupt data in peak files See the ’FileInfo –help’ for details.

1.2.2 Validation of XML files

If you are experiencing problems while processing an XML file you can check if the file does validate against the XML schema using the **FileInfo** tool.

Validation is available for several file formats including MzData, FeatureXML, IdXML.

1.2.3 Converting your files to mzData

The TOPP tools work only on the HUPO-PSI *mzData* format. If you need to convert *mzXML* or *ANDI/MS* data to *mzData*, you can do that using the **FileConverter**, e.g.

```
FileConverter -in infile.mzXML -out outfile.mzData
```

If you use the format names as file extension, the tool derives the format from the extension. For other extensions, the file formats of the input and output file can be given explicitly.

1.2.4 Converting between DTA and mzData

Sequest DTA files can be extracted from a mzData file using the **DTAExtractor:**

```
DTAExtractor -in infile.mzData -out outfile
```

The retention time of a scan, the precursor mass-to-charge ratio (for MS/MS scans) and the file extension are appended to the output file name.

To combine several files (e.g. DTA files) to an mzData file use the **FileMerger:**

```
FileMerger -in infile_list.txt -out outfile.mzData
```

The retention times of the scans can be generated, taken from the *infile_list.txt* or can be extrated from the DTA file names. See the FileMerger documentation for details.

1.2.5 Extracting part of the data from a file

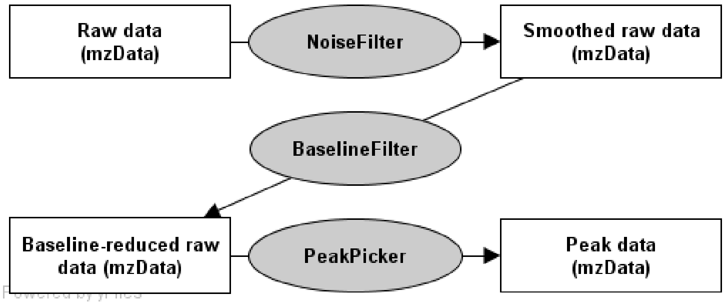
If you want to extract part of the data from an mzData file, you can use the **FileFilter** tool. It allows filtering for RT, m/z and intensity range or for MS level. To extract the MS/MS scans between retention time 100 and 1500, you would use the following command:

```
FileFilter -in infile.mzData -levels 2 -rt 100:1500 -out outfile.mzData
```

1.3 Example 2: Raw data processing

Goal: You want to find all peaks in your raw data.

The first step shown here is the elimination of noise using the NoiseFilter. The now smoothed raw data can be further processed by subtracting the baseline with the BaselineFilter. Then use the PeakPicker to find all peaks in the baseline-reduced raw data.



We offer two different smoothing filters: a Gaussian filter and a Savitzky Golay filter, which can be selected by the option '`type`'. If you want to use the Savitzky Golay filter, or our *BaselineFilter* with non equally spaced raw data, e.g. TOF data, you have to generate equally spaced data by setting the '`resampling`' option.

1.3.1 Finding the right parameters for the NoiseFilter, the BaselineFilter and the PeakPicker

Finding the right parameters is not trivial. The default parameters will not work on most datasets. In order to find good parameters, we propose the following procedure:

1. Load the data in TOPPView
2. Extract a single scan from the middle of the HPLC gradient (Right click on scan)
3. Experiment with the parameters until you have found the proper settings
    - You can find the *NoiseFilter*, the *BaselineFilter*, and the *PeakPicker* in *TOPPView* in the menu 'Layer' - 'Apply TOPP tool'
    - The most important parameters for the *PeakPicker* are
        - `peak_bound` - The minimum intensity of a peak in a MS scan
        - `peak_bound_ms2_level` - The minimum intensity of a peak in a MS/MS scan
        - `fwhm_bound` - The minimal width of a peak
        - `signal_to_noise` - The minimum signal-to-noise ratio a raw data point has to reach in order to be considered a peak. The lower this value is, the more low-intensity peaks will be reported.
        - `scale` - Scale of the wavelet (can be set equal to 'fwhm_bound')

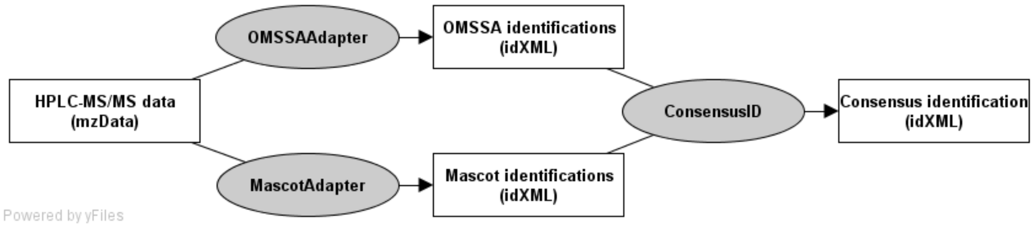
1.4 Example 3: Consensus peptide identification

Goal: Use several identification engines in order to compute a consensus identification for a HPLC-MS\MS experiment.

OpenMS offers adapters for the following commercial and free peptide identification engines: Sequest, Mascot, OMSSA and Inspect.

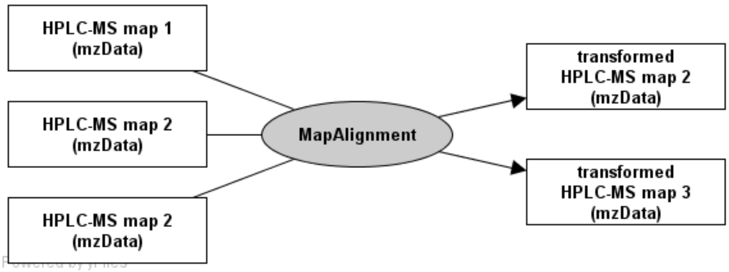
The adapters allow setting the input parameters and data for the identification engine and return the result in the OpenMS idXML format.

In order to improve the identification accuracy, several identification engines can be used and a consensus identification can be calculated from the results. The image below shows an example where Mascot and OMSSA results are fed to the **ConsensusID** tool.

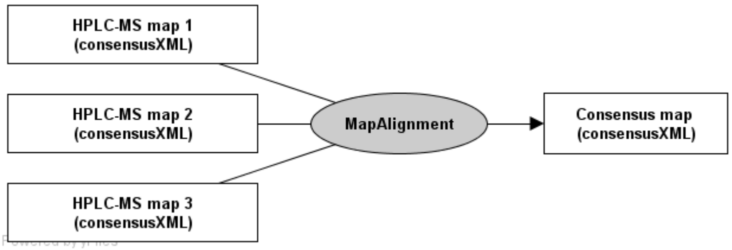


1.5 Example 4: Map alignment

Goal 1: Peak map alignment algorithms should map a number of peak maps onto comparable retention time and m/z dimensions.

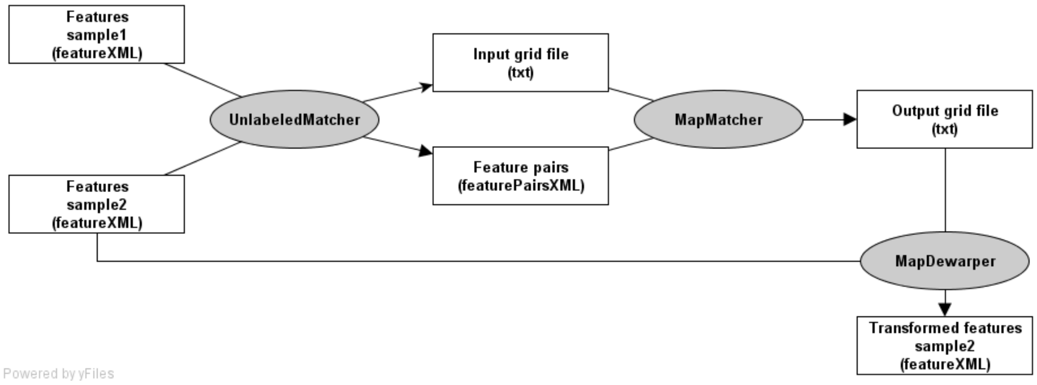


Goal 2: On consensus and feature maps, the alignment approaches should find and group corresponding elements in the maps.


HPLC-MS map 1 (consensusXML)
HPLC-MS map 2 (consensusXML)
HPLC-MS map 3 (consensusXML)
MapAlignment
Consensus map (consensusXML)
Powered by yFiles

The TOPP tool **MapAlignment** solves both alignment problems. Our approach is based on the combination of pairwise map alignments. A pairwise map alignment proceeds in several steps. In the first step, the retention time warp and the distortion in m/z is estimated using a pose clustering approach. This initial transformation is used to find elements in the two maps which likely belong together. In the second step, these pairs are used as landmarks and a final, improved transformation is estimated by which the two maps are mapped onto each other in a third step. The first three steps constitute the so called *superposition phase*. In case of a feature map alignment, the corresponding elements are grouped together in a fourth step, the so called *consensus phase*.

In addition to the *MapAlignment*, we offer three tools for the superposition phase of a pairwise feature map alignment which are **UnlabeledMatcher**, **MapMatcher** and **MapDewarper**. These tools can be used if you want a fine-grained control over the matching process or if you are not interested in the feature pairs, but in the actual mapping function.


Features sample1 (featureXML)
Features sample2 (featureXML)
UnlabeledMatcher
Input grid file (txt)
Feature pairs (featurePairsXML)
MapMatcher
Output grid file (txt)
MapDewarper
Transformed features sample2 (featureXML)
Powered by yFiles

1.5.1 Most important MapAlignment parameters

- `map_type` - The type of the input maps (’peak_map’,’feature_map’, or ’consensus_map’)
- `matching_algorithm:number_buckets` - The number of buckets in retention time and m/z dimension. If the number is set to one, a globally defined warp is estimated and if the number is greater than one, the MapAlignment results in a piecewise defined transformation.
- `matching_algorithm:superimposer:tuple_search:mz_bucket_size` - The deviation of corresponding elements in m/z.
- `matching_algorithm:pairfinder:precision` - The deviation of corresponding elements in retention time and m/z after dewarping.

1.6 Example 5: Quantitation

1.6.1 General introduction to the FeatureFinder

For quantitation, the *FeatureFinder* tool is used. It extracts the features from raw or peak maps. The *FeatureFinder* offers different algorithms:

Algorithm	Input data	Description
simplest	raw data	This is an algorithm for feature detection in raw data. The following components are used: **Seeding:** A "seed" is a starting point for finding a feature. In this algorithm, all raw data points above the noise threshold are sorted with respect to intensity. The strongest unused data point is used as a seed. (class SimpleSeeder) **Extension:** A "region" is extended around the seed. The region grows in RT and m/z simultaneously. These data points are marked as used. (class SimpleExtender) **Modelling:** A theoretical "model" is fitted to the data points of the region. Both dimensions (RT, m/z) are considered separately. This algorithm uses a bi-Gaussian model for the elution profile, i.e., two "half" Gaussians are chosen to represent the data points to the left and right of the maximum intensity. This is done by maximum likelihood estimation (class BiGaussModel). The m/z dimension is represented by an isotope model, which is essentially a mixture of Gaussians, one for each isotopic peak, whose relative intensities are fixed according to the "averagine" atomic composition. All isotopic peaks have the same width. As a special case, charge zero represents a single Gaussian. This is useful as a null hypothesis or if isotopic peaks cannot be resolved. The MZ model is fitted by a simple enumeration scheme. (class IsotopeFitter1D) If the model fits well to the data, we report a feature. Otherwise, the seed and the whole region is discarded, and its data points are marked as unused again. (class ModelFitter) See the `FeatureFinderAlgorithmSimplest Parameters` page for a documented list of configuration options for this

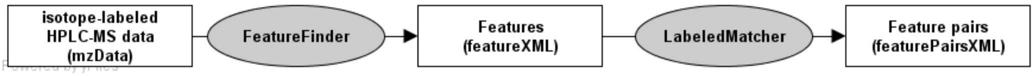
1.6.2 Isotope-labeled quantitation

Goal: You want to differentially quantify the features of an isotope-labeled HPLC-MS map.

The first step in this pipeline is to find the features of the HPLC-MS map. The FeatureFinder application calculates the features from a raw/peak map.

In the second step, the labeled pairs (light/heavy) are determined by the LabeledMatcher. The LabeledMatcher first determines all possible pairs according to a given optimal shift and deviations in RT and m/z. Then it resolves ambiguous pairs using a greedy-algorithm that prefers pairs with a higher score. The score of a pair is the product of:

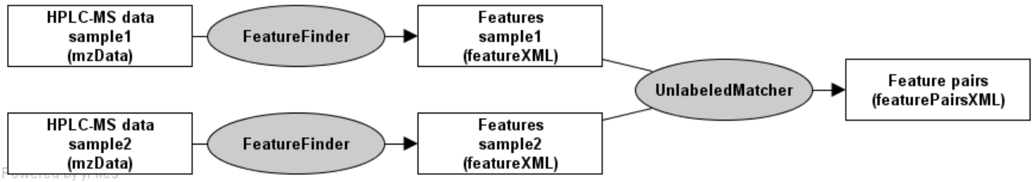
- feature quality of feature 1
- feature quality of feature 2
- quality measure for the shift (how near is it to the optimal shift)


isotope-labeled HPLC-MS data (mzData)
FeatureFinder
Features (featureXML)
LabeledMatcher
Feature pairs (featurePairsXML)

1.6.3 Label-free quantitation

Goal: You want to differentially quantify the features of two or more label-free HPLC-MS map.

Mapping feature maps can be done with the *MapAlignment* tool. Please have a look at Example 4: Map alignment.


HPLC-MS data sample1 (mzData)
FeatureFinder
Features sample1 (featureXML)
HPLC-MS data sample2 (mzData)
FeatureFinder
Features sample2 (featureXML)
UnlabeledMatcher
Feature pairs (featurePairsXML)

1.6.4 References

Ole Schulz-Trieglaff, Rene Hussong, Clemens Grï£¡pl, Andreas Leinenbach, Andreas Hildebrandt, Christian Huber, Knut Reinert "Computational Quantification of Peptides from LC-MS data". Journal of Comptational Biology, 2008. to appear.

Ole Schulz-Trieglaff, Rene Hussong, Clemens Grï£¡pl, Andreas Hildebrandt, Knut Reinert "A Fast and Accurate Algorithm for the Quantification of Peptides from Mass Spectrometry data". In "Proceedings of the Eleventh Annual International Conference on Research in Computational Molecular Biology (RECOMB 2007)", pages 473-487, 2007.

Bettina Mayr, Oliver Kohlbacher, Knut Reinert, Marc Sturm, Clemens Grï£¡pl, Eva Lange, Christoph Klein, Christian Huber "Absolute Myoglobin Quantitation in Serum by Combining Two-Dimensional Liquid Chromatography-Electrospray Ionization Mass Spectrometry and Novel Data Analysis Algorithms". Journal of Proteome Research, volume 5, pages 414-421, 2006.

Clemens Grï£¡pl, Eva Lange, Knut Reinert, Oliver Kohlbacher, Marc Sturm, Christian G. Huber, Bettina M. Mayr, Christoph L. Klein "Algorithms for the automated absolute quantification of diagnostic markers in complex proteomics samples". In "Proceedings of the 1st International Symposium on Computational Life Science (CompLife05)", pages 151-163, 2005.

1.7 Example 6: Calibration

1.7.1 General introduction to the calibration

We offer two calibration methods: an internal and an external calibration. Both can handle peak data as well as raw data. If you want to calibrate raw data, a peak picking step is necessary, the important parameters can be set via the ini-file. If you have already picked data, don't forget the '`-peak_data`' flag.

The external calibration is used to convert flight times into m/z- values with the help of external calibrant spectra containing e.g. a polymer like polylysine. For the calibrant spectra, the calibration constants the machine uses need to be known as well as the expected masses. Then a quadratic function is fitted to convert the flight times into m/z-values.

The internal calibration uses reference masses in the spectra to correct the m/z-values using a linear function.

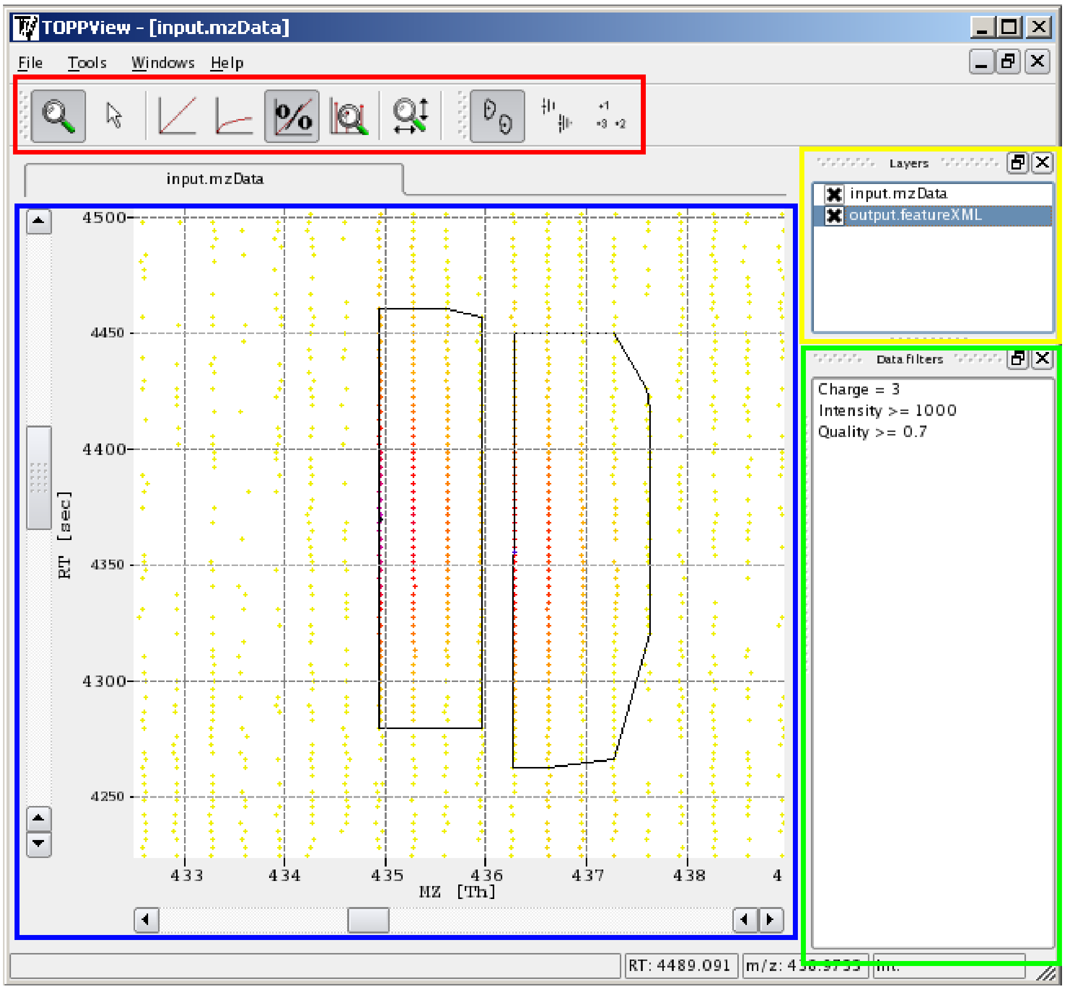
2 TOPPView Tutorial

2.1 Introduction

TOPPView is a viewer for MS and HPLC-MS data. It can be used to inspect files in mzData, mzXML, ANDI/MS and several other text-based file formats. It also supports viewing data from an OpenMS database.

In each view, several datasets can be displayed using the layer concept. This allows visual comparison of several datasets as well as displaying input data and output data of an algorithm together.

TOPPView is intended for visual inspection of the data by experimentalists as well as for analysis software by developers.



Charge = 3
Intensity >= 1000
Quality >= 0.7

The above example image shows a 2D view of TOPPView (blue rectangle) and the corresponding Display and action modes (red rectangle). In the right dock area, you can see the Layers (yellow rectangle) and the Data filtering tool.

2.1.1 Layers

Each view of TOPPView supports several datasets, called layers. In the *layer manager*, dock window, layers can be hidden and shown using the checkbox in front of each layer name.

By clicking on a layer, this layer is selected, which is indicated by a blue background. The selected layer can be manipulated using the *Tools* menu.

Note:

> Opening two or more datasets in different layers of one window is possible from the command line, too. Execute *TOPPView –help* for more information.

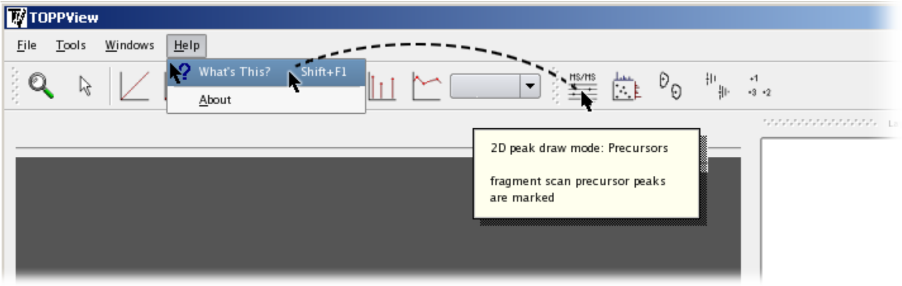
2.1.2 Data filtering

TOPPView allows filtering of the displayed peak data and feature data. Peak data can be filtered according to intensity and metadata. Metadata is arbitrary data the peak is annotated with. Feature data can be filtered according to intensity, charge, quality and metadata.

Data filters are managed by a dock window. Filters can be added, removed and edited through the context menu of the data filters window. For convenience, filters can also be edited by double-clicking them.

2.1.3 Looking for help?

You can display a short help text for each button and dock window of TOPPView by clicking on it in *What's this*? mode. *What's this*? mode can be entered using the *Help* menu.



2D peak draw mode: Precursors

fragment scan precursor peaks are marked

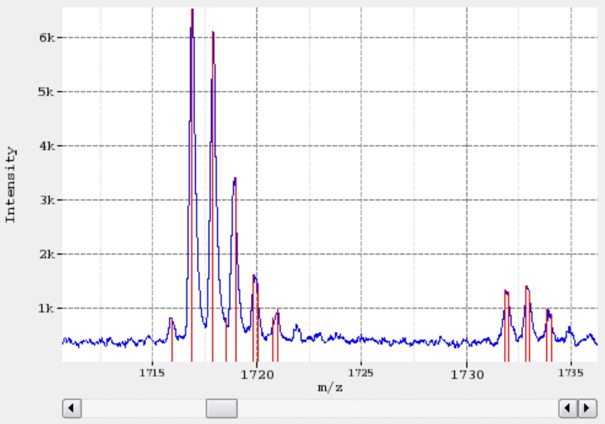
2.2 The views

TOPPView offers three types of views – a 1D view for spectra, a 2D view for peak maps and feature maps, and a 3D view for peak maps. All three views support zooming for a closer look. They can be freely configured to suit the individual needs of the user.

2.2.1 1D view

The 1D view is used to display raw spectra or peak spectra. Raw data is displayed using a continuous line. Peak data is displayed using one stick per peak. The color used for drawing the lines can be set for each layer individually.

The following example image shows a raw data spectrum and the corresponding peak spectrum:



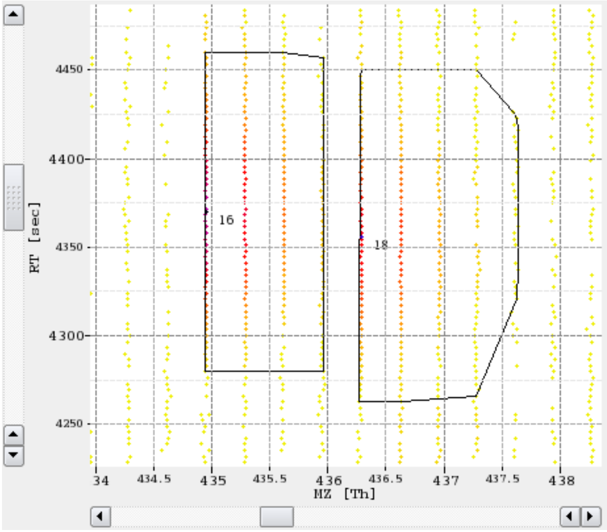
Context menu: Through the context menu of the 1D view you can:

- Save the current layer data
- Change display settings

2.2.2 2D view

The 2D view is used to display peak maps and feature maps in a top-down view with color-coded intensities. Peaks and feature centroids are displayed as dots. For features, also the overall convex hull and the convex hulls of individual mass traces can be displayed. The color gradient used to encode for peak and feature intensities can be set for each layer individually.

The following example image shows a small section of a peak map and the detected features in a second layer.



In addition to the normal top-down view, the 2D view can display the projections of the data to the m/z and RT axis. This feature is mainly used to assess the quality of a feature without opening the data region in 3D view.

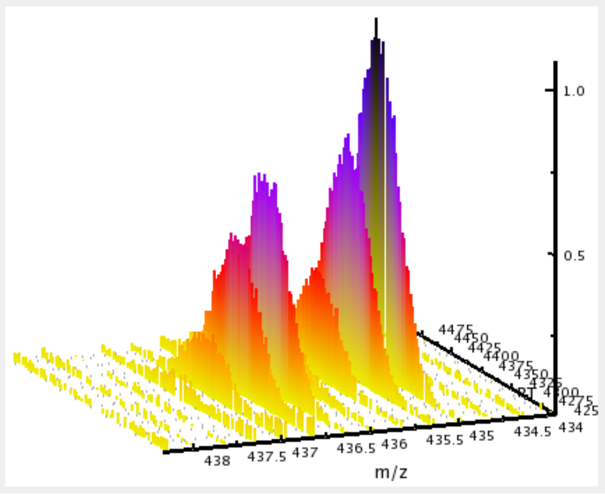
Context menu: Through the context menu of the 2D view you can:

- View survey/fragment scans in 1D view
- View survey/fragment scans meta data
- View the currently selected area in 3D view
- Save the current layer data
- Change display settings

2.2.3 3D view

The 3D view can display peak maps only. Its primary use is the closer inspection of a small region of the map, e.g. a single feature. In the 3D view slight intensity differences are easier to recognize than in the 2D view. The color gradient used to encode peak intensities, the width of the lines and the coloring mode of the peaks can be set for each layer individually.

The following example image shows a small region of a peak map:



Context menu: Through the context menu of the 3D view you can:

- Save the current layer data
- Change display settings

2.3 Display and action modes

All of the views support several display modes and several action modes. Display modes determine how intensities are displayed. Action modes allow different types of interaction with the data.

2.3.1 Display modes

Intensity display modes determine the way peak intensities are displayed.

- **Linear:** Normal display mode.
- **Percentage:** In this display mode the intensities of each dataset are normalized with the maximum intensity of the dataset. This is especially useful in order to visualize several datasets that have large intensity differences. If only one dataset is opened it corresponds to the normal mode.
- **Snap to maximum intensity:** In this mode the maximum currently displayed intensity is treated as if it was the maximum overall intensity.

2.3.2 Action modes

Action modes determine the mouse actions. Action modes not supported in the chosen spectrum display mode are displayed in gray.

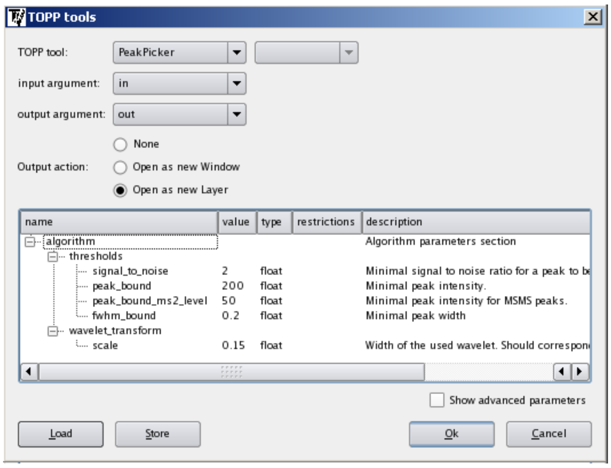
- **Zoom + Translate:** Allows zooming to a specific data area. By pressing the CTRL key you can translate the displayed area.
- **Select + Measure:** The m/z, RT and intensity of a selected peak are displayed in the status bar in this mode. By pressing the CTRL key you can determine the difference in m/z and RT, and the intensity ratio of the selected peaks.

2.4 Data analysis

TOPPView also offers limited data analysis capabilities for single layers, which will be illustrated in the following sections. The functionality presented here can be found in the *Tools* menu.

2.4.1 TOPP tools

Single TOPP tools can be applied to the data of the currently selected layer. The following example image shows the TOPP tool dialog:



TOPP tool: PeakPicker

input argument: in

output argument: out

Output action:
○ None
○ Open as new Window
● Open as new Layer

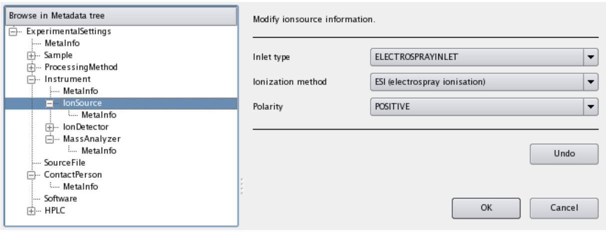
name	value	type	restrictions	description
algorithm				Algorithm parameters section
thresholds				
signal_to_noise	2	float		Minimal signal to noise ratio for a peak to be
peak_bound	200	float		Minimal peak intensity.
peak_bound_ms2_level	50	float		Minimal peak intensity for MSMS peaks.
fwhm_bound	0.2	float		Minimal peak width
wavelet_transform				
scale	0.15	float		Width of the used wavelet. Should correspon

☐ Show advanced parameters

After selecting a tool, the algorithm parameters can be edited manually, or loaded from an INI file. The results can then be displayed as a new layer of the same window or in a new window.

2.4.2 Metadata

One can access the metadata the layer is annotated with. This data comprises e.g. contact person, instrument description and sample description.



Modify ionsource information.

Inlet type: ELECTROSPRAYINLET

Ionization method: ESI (electrospray ionisation)

Polarity: POSITIVE

Note:

> Identification data, e.g. from a Mascot run, can be annotated to the spectra or features, too. After annotation, this data is listed in the metadata as well.

2.4.3 Statistics

Statistics e.g. about peak/feature intensities can be displayed. For intensities, it is possible to display an additional histogram view.