



# SPECTROview User Manual

(version 15/01/2026)

## Table of contents

<b>1. Introduction .....</b>	<b>2</b>
<b>2. Installation .....</b>	<b>2</b>
<b>3. Supported data by SPECTROview .....</b>	<b>3</b>
3.1 Spectroscopic data (.txt, .csv).....	3
3.2 Hyperspectral data (.txt, .csv) .....	3
3.3 Datasheet (Excel files).....	3
3.4 Formats saved by SPECTROview application .....	4
<b>4. User Interface Overview .....</b>	<b>4</b>
<b>5. Spectra and Maps workspaces .....</b>	<b>5</b>
5.1 SpectraList and MapList .....	6
5.2 Spectra Viewer.....	7
5.3 Fit Model Builder .....	9
5.3.1 Fitting Panel.....	9
5.3.2 PeakTable Panel .....	10
5.3.3 FitModelControl Panel .....	11
5.4 Collect and Save Fit Results.....	11
5.5 Save workspace.....	12
<b>6. « Graphs » workspace .....</b>	<b>12</b>
6.1 Loading data .....	13
6.2 Add a new plot .....	13
6.3 Update/Modify existing plot.....	13
6.4 Data filtering feature.....	14
6.5 Other plot customization .....	14



## 1. Introduction

Spectroscopy techniques (such as Raman, Photoluminescences, XRD, XPS...) are widely used in various fields, including materials science, chemistry, biology, and geology. In recent years, these techniques have increasingly found their place in cleanroom environments, particularly within the microelectronics industry, where they serve as critical metrology tools for wafer-scale measurements. The data collected from these in-line measurements (wafer data) require specific processing, but existing software solutions are often not optimized for this type of data and typically lack advanced plotting and visualization capabilities. Additionally, the licensing requirements of these software solutions can restrict access for a broader community of users.

SPECTROview addresses these gaps by offering free, open-source software that is compatible with both in-line data (wafer-map) as well as standard spectroscopic data (discrete spectra, 2D maps). It also features a built-in visualization tool, enabling users to streamline both data processing and visualization in a single application, making the workflow more efficient.

More details regarding the latest updates and features of the SPECTROview can be found in the Github repository : <https://github.com/CEA-MetroCarac/SPECTROview>

Dont forget to cite the tool when it is used for your data processing , data visualization on your publication:

*Le, V.-H., & Quémésré, P. (2025). SPECTROview : A Tool for Spectroscopic Data Processing and Visualization. Zenodo. <https://doi.org/10.5281/zenodo.14147172>*

### Highlighted features:

- Cross-platform compatibility (Windows, macOS, Linux).
- Supports processing of spectral data (1D) and hyperspectral data (2D maps or wafer maps).
- Ability to fit multiple spectra or 2Dmaps using predefined custom fit models.
- Collect all best-fit results with one click.
- Optimized user interface for easy and quick inspection and comparison of spectra.
- Dedicated module for effortless, fast, and easy data visualization.
- Each module will be described in detail in the following sections.

## 2. Installation

Ensure that Python (version between v3.8 and v3.12) is already installed on your PC.

The installation of SPECTROview can be performed via the command line (CMD):

```
pip install spectroview
```

You can also install directly via the GitHub repository by copy and paste following line in CMD:

```
pip install git+https://github.com/CEA-MetroCarac/SPECTROview.git
```

To open SPECTROview, Open CMD then type:

```
spectroview
```

To update SPECTROview to the newest version:

```
pip install --upgrade spectroview
```

To update SPECTROview to the specific version:

```
pip install --upgrade spectroview==num-of-version
```

Example: `pip install --upgrade spectroview==26.3.1`



### 3. Supported data by SPECTROview

Examples of all supported data types can be found in the `/examples` folder within the [GitHub repository](https://github.com/CEA-MetroCarac/SPECTROview/tree/main/examples):

<https://github.com/CEA-MetroCarac/SPECTROview/tree/main/examples> .

Users can download these files to understand the supported files format and data structures; practice using the SPECTROview application with real examples.

#### 3.1 Spectroscopic data (.txt, .csv)

SPECTROview supports spectroscopic data (TXT or CSV format). The files must consist of two columns, separated by *semicolon* or *space*, or *tab*. Data files can contain or not column headers:

- **Column 1:** Raman shift ( $\text{cm}^{-1}$ )
- **Column 2:** Corresponding Raman intensity (a.u.)

header	header
$R_1$	Intensity
$R_2$	Intensity
...	...
$R_{n-1}$	Intensity
$R_n$	Intensity

#### 3.2 Hyperspectral data (.txt, .csv)

SPECTROview supports hyperspectral data (2D maps or wafer maps) in TXT or CSV format, with the data structure arranged as follows:

		$R_1$	$R_2$	$R_3$	...	$R_{n-1}$	$R_n$
<b>X1</b>	<b>Y1</b>	$I_1$	$I_1$	$I_1$	...	$I_1$	$I_1$
<b>X2</b>	<b>Y2</b>	$I_2$	$I_2$	$I_2$	...	$I_2$	$I_2$
...	...	...	...	...	...	...	...
<b>Xn</b>	<b>Yn</b>	$I_n$	$I_n$	$I_n$	...	$I_n$	$I_n$

- The first row ( $R_1 \rightarrow R_n$ ) is the Raman shift ( $\text{cm}^{-1}$ ) of all spectrum
- Two first columns listing the **X** and **Y** coordinate of the spectrum within the maps.
- The remaining columns contain the corresponding **intensity** values for each spectrum ( $I_1 \rightarrow I_n$ ), from the second row to the last.

**Note:** 2D map formats from Renishaw tools must be converted before they can be used in SPECTROview. An integrated conversion tool is provided in SPECTROview application for this purpose (see Section 4, File Convert Tool)

#### 3.3 Datasheet (Excel files)

Excel files (.xlsx, .xls) containing one or multiple sheets can be directly loaded into SPECTROview.



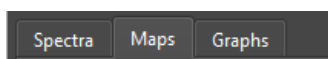
### 3.4 Formats saved by SPECTROview application

Depending on the active workspace, SPECTROview saves files with specific extensions so work can be resumed later:

- Maps workspace → .maps
- Spectra workspace → .spectra
- Graphs workspace → .graphs

## 4. User Interface Overview

The SPECTROview application is designed for the efficient processing of spectroscopic data and the easy visualization of fitted results. The interface features three main workspaces, each developed for a specific purpose:



- **Spectra**: For processing one or multiple discrete spectra.
- **Maps**: For processing one or multiple hyperspectral datasets, including wafer data and 2D maps.
- **Graphs**: For plotting and visualizing data.

**Tooltip**: Most GUI elements in SPECTROview (buttons, text boxes, dropdowns, etc.) feature tooltips. → To see a brief explanation of its function: hover the mouse cursor over an element for 1 second.

**Toolbar**: a horizontal toolbar is located at the top edge of the application containing the following buttons:

	<b>Loads</b> all supported data types as listed in section 3. The application will automatically switch to the appropriate workspace based on the loaded file type.
	<b>Saves</b> the current active workspace to a file, allowing the user to reopen it later and resumes the work.  As detailed in Section 3.4, the file extension will be .maps, .spectra, or .graphs depending on the active workspace.
	<b>Clears</b> the current active workspace to start a new session. All loaded data will be removed.
	<p>Opens the File Convert Tool to convert hyperspectral data (2D maps) from the Renishaw WiRE format into a format supported by SPECTROview:</p> <p><u>How to use (see Figure 1):</u></p> <p>Load the file(s) to convert → Selected file(s) to convert from the list (4.2) → Click button “Convert” (4.3) → The new file is created in the same folder with the suffix <u>_converted</u> (4.4).</p>

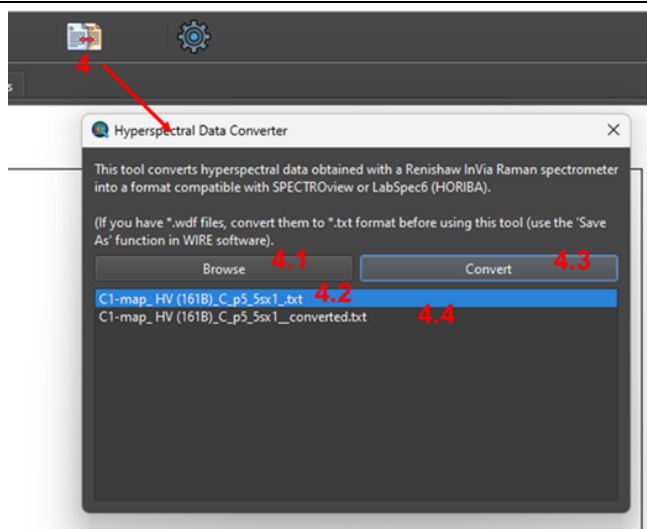


Figure 1: Interface of “Hyperspectral Data Converter”



	<p><b>“Settings”</b>: Opens the <b>Settings Panel</b> to adjust fitting parameters and define the storage folder for user-defined fitting models</p>	
	<p><b>Theme Toggle</b>: Switches the application GUI between Dark and Light modes.</p>	
	<p>To Open this <b>User Manual</b> document.</p>	
	<p>To displays version information and details about the application..</p>	

Figure 2: Setting Panel

## 5. Spectra and Maps workspaces

The **Maps** and **Spectra** workspaces are very similar. They share many common features, with the Maps workspace containing additional specific features designed to handle multiple hyperspectral datasets (2D maps or wafer maps). The following section details the GUI of the Maps workspace.

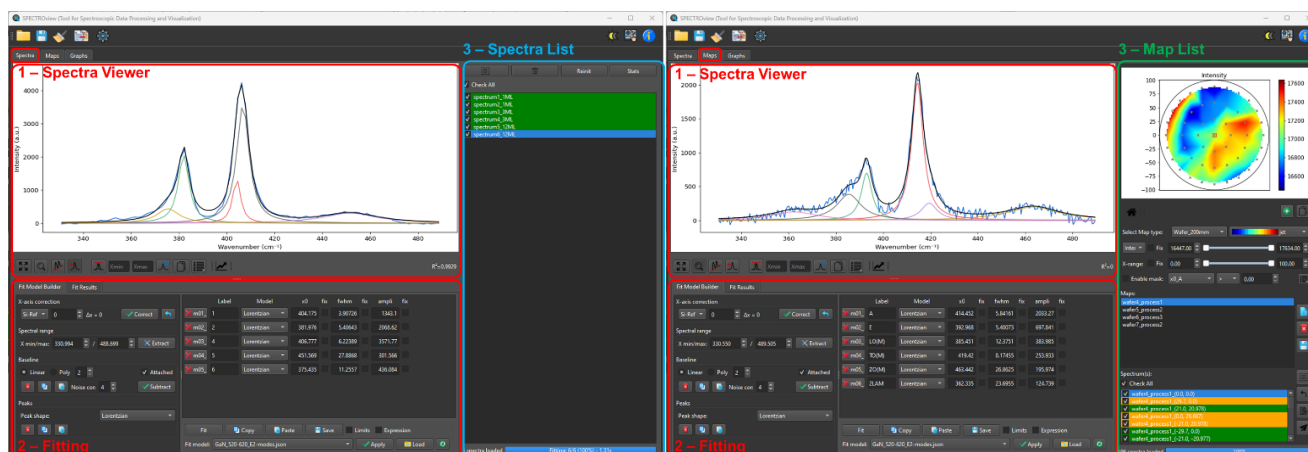


Figure 3: Interface overview of **Spectra** (left) and **Maps** (right) workspaces. The interface is divided into three main sections. Section **1-SpectraViewer** and **2-FittingPanel** are identical in both workspaces, whereas section **3-SpectraList** (in **Spectra** workspace) or **3-MapList** (in **Maps** Workspace) are different.



## 5.1 SpectraList and MapList

As showed in Figure 3, the **SpectraList** and **MapList** sections are designed differently for **Spectra** and **Map** workspaces in order to efficiently navigate between spectral or hyperspectral (maps) data.

In the following, the **MapList** section of the **Maps** workspace (which also contains also a **SpectraList** module) is described in detail (Figure 4).

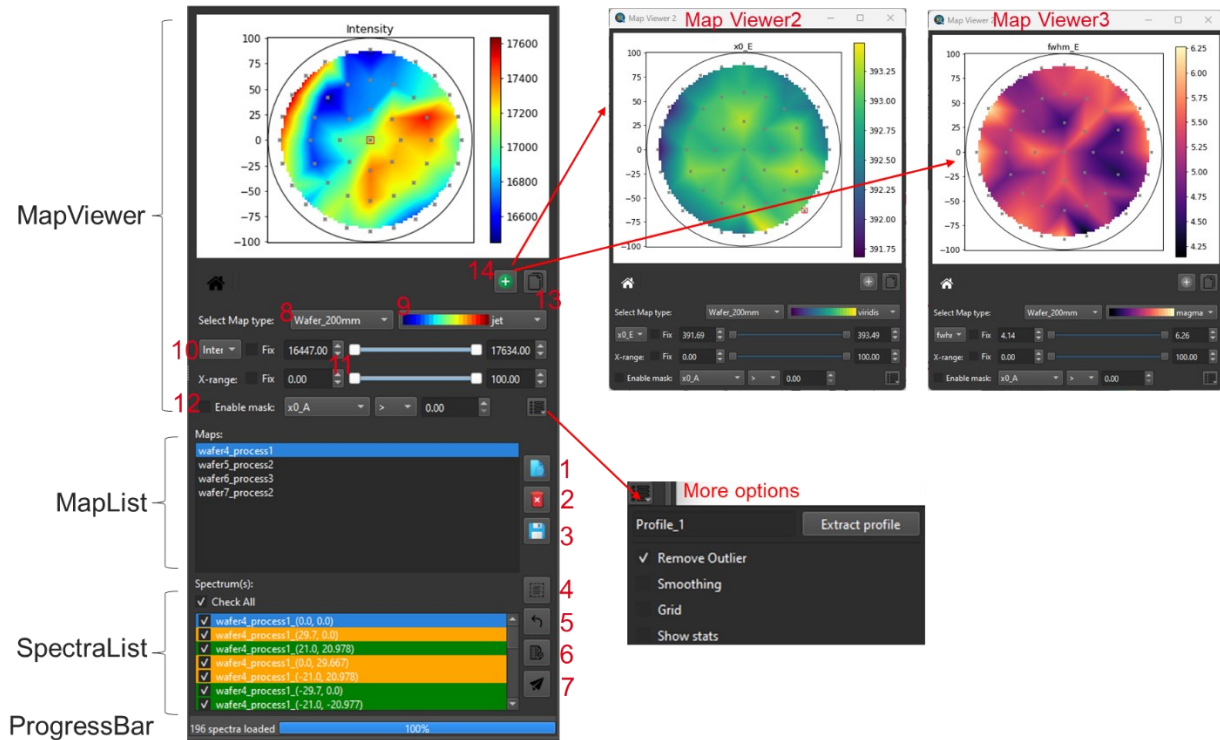


Figure 4: (Left) **MapList** section within the **Maps** workspace. On the right: (top) additional **MapView** module(s) can be added as floating window(s); (bottom) **More Options** to adjust and customize the heatmap display.

**MapList:** The MapList displays all loaded map files, including wafer maps and 2D map types. Three buttons located on the right side of the MapList allow the user to (1) **view** the selected map data, (2) **delete** the selected map, or (3) **save** the selected map to an Excel file (see Figure 4).

**SpectraList:** all loaded spectra in the Spectra workspace, or all spectra associated with the map selected in the MapList (in the Maps workspace). Users can select one or multiple spectra simultaneously. The selected spectra are displayed in the SpectraViewer (cf. section 5.2). Four buttons associated with the SpectraList allow the user to (4) **Selected all available spectra** in the list, (5) **reset** (reinit) the selected spectra, (6) **display the fit statistic report** or (7) **send the selected spectra to the Spectra workspace** (see Figure 4).

**MapView:** displays the heatmap of the selected map (wafer or 2D map type) chosen from the **MapList**. Several options are available to customize the heatmap:

- (8) **Map type:** 2D map or wafer map with different diameters.
- (9) **Color palette** selection for the heatmap.
- (10) **Displayed parameter:** maximum intensity, area, or any fitted parameter.
- (11) **Range sliders:** two sliders are available to adjust the X-axis range and the color bar limits.
- (12) **Mask feature:** allows the user to define specific regions of the heatmap based on user-defined filters.
- (13) **Copy button:** copies the heatmap to the clipboard, allowing it to be pasted elsewhere.



- **(14) Multiple MapViewers:** users can add additional MapViewer windows as floating windows, enabling comparison between different fitted parameters of the same map (see Figure 4) (see Figure 4).

**ProgressBar:** placed under the SpectraList (see Figure 5) displays the number of spectra available in the SpectraList, and the progress of the fitting process (number of fitted spectra / total spectra, percentage (%), and elapsed time (s)). The Stop button allow to stop the fitting process at anytime.

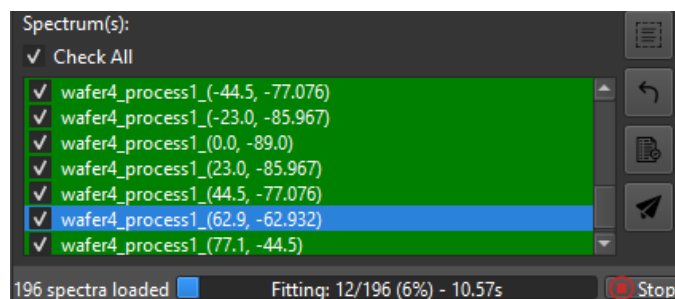


Figure 5: SpectraList and ProgressBar

## 5.2 Spectra Viewer

SpectraViewer plot all spectra (and it best fit curves) selected via the SpectraList (cf. section 5.1)

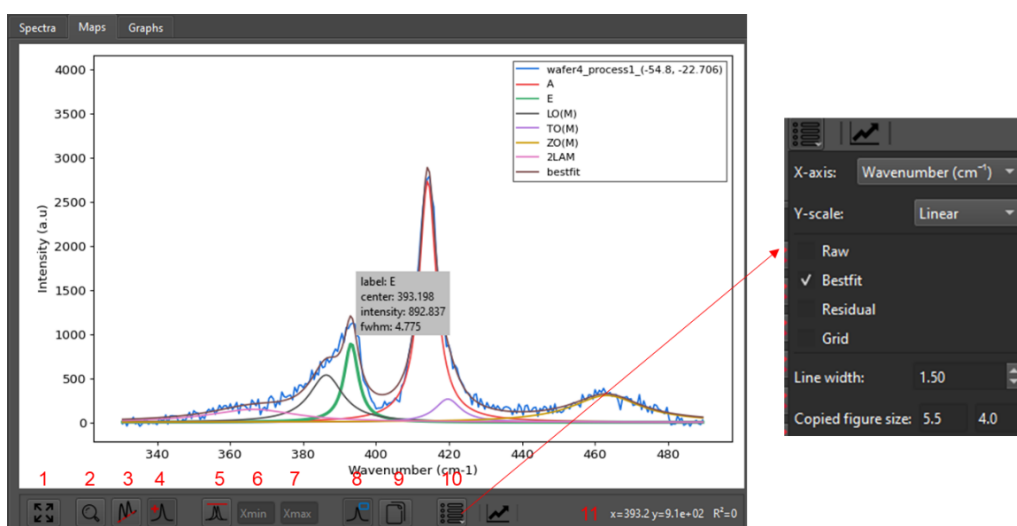


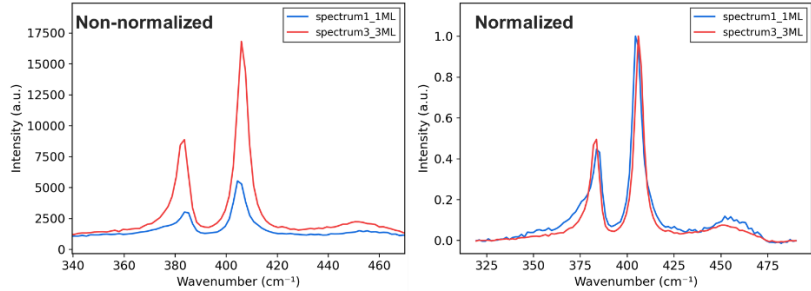

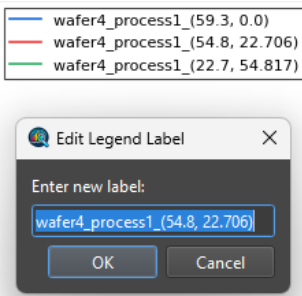


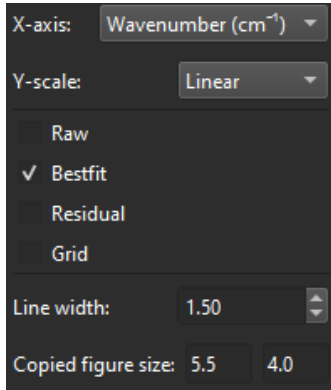


Figure 6: Spectra Viewer (left) widget and View Options Menu (right).


	<b>Rescale:</b> to rescale the spectra plot. Shortcut: Ctrl + R
	<b>Zoom:</b> when active, enables the "zoom" feature of spectra plot using the left mouse click & drag
	<b>Baseline:</b> when active, allows the user to define baseline point(s) using the left mouse click.
	<b>Peaks:</b> when active, allows the user to define peak(s) on the spectra using the left mouse click.
	<b>Normalization:</b> when clicked, displays all selected spectra normalized to the maximum peak intensity. If the user wants to normalize to a specific spectra range or specific peak → type the corresponding spectral range into text box 'min' and 'max' (see Figure 7).





 	<div data-bbox="427 165 1241 454">  </div> <p data-bbox="256 472 1422 539">Figure 7: Example of <b>Raw</b> spectra (left) and spectra <b>normalized</b> to the maximum intensity within the 400-420 cm<sup>-1</sup> range (right) to facilitate the inspection of peak shifts.</p>
	<p data-bbox="240 575 925 609"><b>Legend:</b> displays the legend for all selected spectra.</p> <p data-bbox="240 624 1011 725">Once the legend is shown, the user can click directly on the legend box to change the color or labels (see figure on the right).</p> <p data-bbox="240 741 1011 808"><b>Note:</b> This feature only works when the <b>Zoom</b> button is disabled (by selecting <b>Baseline</b> or <b>Peak</b> button).</p> <div data-bbox="1082 575 1385 869">  </div>
	<p data-bbox="240 900 1434 967"><b>Copy:</b> copies the spectra plot to the clipboard as an image. The user can then paste it into other applications (PowerPoint, Excel, etc.).</p> <ul data-bbox="292 983 1434 1050" style="list-style-type: none"> <li>- <b>Ctrl + Click</b> (or <b>Cmd + click</b> on macOS): Copies the numerical data of the spectra (including RAW data and best-fit curves) to the clipboard.</li> </ul>
	<p data-bbox="240 1075 1050 1142"><b>More options:</b> opens a pop-up window with several view options (cf. figure on the right).</p> <ul data-bbox="252 1158 1031 1529" style="list-style-type: none"> <li>- <b>X-axis unit</b> : available option (Wave number (cm<sup>-1</sup>), wavelength (nm), Emission energy (eV), binding energy (eV), Frequency (Hz), 2Theta (degree)).</li> <li>- <b>Y axis-scale</b>: <i>Linear or log scale</i>.</li> <li>- <b>Raw</b> : Toggles the visibility of the raw data.</li> <li>- <b>Besfit</b>: Toggles the visibility of the fitted peaks.</li> <li>- <b>Residual</b>: Displays the residual (Raw - bestfit).</li> <li>- <b>Grid</b>: adding grids on the plot.</li> <li>- <b>Line width</b>: to adjust the line with of the spectra plot.</li> <li>- <b>Figure size</b>: to adjust the ratio (high-width) of copied figure.</li> </ul> <div data-bbox="1086 1070 1418 1458">  </div>

### Other features of SpectraViewer:

- **Show Peak Parameters** on the spectra plot: by hovering the mouse cursor over a peak displays a pop-up with the peak parameters (peak label, intensity, FWHM, position), as shown in Figure 6.
- **Add/Remove Peaks**: Right-click to remove the current peak, or left-click to add a new peak (ensure the Peak button  is active).
- **Adjust peak by dragging**.
- **Quick re-scale** of the y axis: Use the mouse wheel to quickly adjust the Y-axis (intensity scale).





### 5.3 Fit Model Builder

The FitModelBuilder Tab widget contains three main panel : Fitting, PeakTable Panel, FitModelControl Panels (Figure 8).

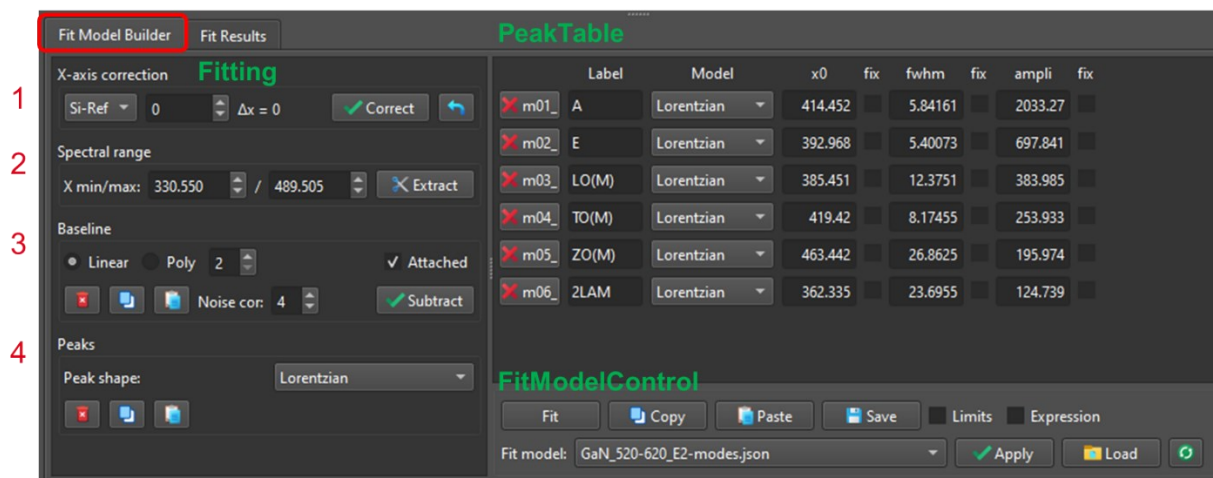


Figure 8: FitModelBuilder tab widget containing three main panels: (1) Fitting, (2) PeakTable and (3) FitModelControl.

#### 5.3.1 Fitting Panel

The FittingPanel (left panel of Figure 8) is divided four small sections, each corresponding to the key steps in the fitting process or in the construction of the fit model.

**Step 1: x-axis Correction (optional)** : This step is used to perform an x-axis correction based on measurements from a well-known reference sample. In SPECTROview, this feature is currently implemented for a **silicon reference sample**, which has a theoretical Raman peak position at  $520.7 \text{ cm}^{-1}$ . Support for additional reference samples will be added in future versions.

##### Procedure:

- Record the silicon reference spectrum during the same experimental session as the other samples, using identical measurement parameters.
- Fit the Si-Ref spectrum to determine the measured silicon peak position.
- Enter the measured position into the text box.
- Select one or more spectra to correct → the x-axis correction is applied based on the difference between the measured position and the theoretical value ( $520.7 \text{ cm}^{-1}$ ).

**Step 2: Define the fitting range** : to define the **x-axis range** to be used for the fitting process.

##### Step 3: Baseline definition:


- **Baseline points** are defined directly in the spectra plot (SpectraViewer widget) by left-clicking.

The Baseline button  must be active (cf. section 0).

- **Attached checkbox** : When checked, the baseline points remain attached to the spectrum curve.
- **Correct noise**: Calculates each baseline point as the average of the specified number of neighboring data points, ensuring it sits in the center of the noise level.
- **Baseline properties**: The baseline type (e.g., linear, polynomial) can be adjusted as needed.

##### Step 4: Peak(s) definition:



- Peaks can be defined directly on the spectra using the Spectra Viewer by left-clicking. The **Peak** button  must be active (cf. section 0).
- Any peak added to the fit model is automatically displayed in the **PeakTable** panel (see below). User can then modify/adjust directly in the PeakTable panel.

### 5.3.2 PeakTable Panel

The **PeakTable** displays all available peaks defined of the selected spectrum.

Peak properties (*label*, *model*, *position*, *FWHM*, *intensity*) are dynamically updated as user modifying via SpectraViewer plot (with mouse *click* or *dragging*). Users can also remove peak(s) or manually adjust their parameters directly via **PeakTable**.

	Label	Model	X0	fix X0	Fwhm	fix Fwhm	Ampli	fix Ampli
<input checked="" type="checkbox"/> m01_	A	Lorentzian	414.258	<input type="checkbox"/>	5.888	<input type="checkbox"/>	2709.714	<input type="checkbox"/>
<input checked="" type="checkbox"/> m02_	E	Lorentzian	392.946	<input type="checkbox"/>	5.045	<input type="checkbox"/>	970.902	<input type="checkbox"/>
<input checked="" type="checkbox"/> m03_	LO(M)	Lorentzian	386.016	<input type="checkbox"/>	11.105	<input type="checkbox"/>	592.371	<input type="checkbox"/>
<input checked="" type="checkbox"/> m04_	TO(M)	Lorentzian	418.887	<input type="checkbox"/>	8.277	<input type="checkbox"/>	345.69	<input type="checkbox"/>
<input checked="" type="checkbox"/> m05_	ZO(M)	Lorentzian	463.469	<input type="checkbox"/>	25.345	<input type="checkbox"/>	301.522	<input type="checkbox"/>
<input checked="" type="checkbox"/> m06_	2LAM	Lorentzian	364.236	<input type="checkbox"/>	25.078	<input type="checkbox"/>	164.159	<input type="checkbox"/>

Figure 9: **PeakTable** panel

**Constraints:** For each peak, one or more parameters can be fixed by selecting the corresponding checkbox. For more advanced constraints, two additional options (via checkboxes **Limit** or **Expression**) are proposed:

- **Limits** checkbox: When enabled, **min** and **max** value columns are displayed for each parameter, allowing the user to restrict parameter variations within a defined range (as opposed to the **Fix** option, which keeps the value constant).

	Label	Model	min X0	X0	max X0	fix X0	min Fwhm	Fwhm	max Fwhm	fix Fwhm
<input checked="" type="checkbox"/> m01_	A	Lorentzian	393.68	414.258	433.68	<input type="checkbox"/>	0	5.888	200	<input type="checkbox"/>
<input checked="" type="checkbox"/> m02_	E	Lorentzian	369.29	392.946	409.29	<input type="checkbox"/>	0	5.045	200	<input type="checkbox"/>
<input checked="" type="checkbox"/> m03_	LO(M)	Lorentzian	360.17	386.016	400.17	<input type="checkbox"/>	0	11.105	200	<input type="checkbox"/>
<input checked="" type="checkbox"/> m04_	TO(M)	Lorentzian	398.61	418.887	438.61	<input type="checkbox"/>	0	8.277	200	<input type="checkbox"/>
<input checked="" type="checkbox"/> m05_	ZO(M)	Lorentzian	437.78	463.469	477.78	<input type="checkbox"/>	0	25.345	200	<input type="checkbox"/>
<input checked="" type="checkbox"/> m06_	2LAM	Lorentzian	338.74	364.236	378.74	<input type="checkbox"/>	0	25.078	200	<input type="checkbox"/>

- **Expression** checkbox: when enabled, **expression** column it allows parameters from different peaks to be linked through mathematical expressions.
  - o Example 1: To constrain the position of peak **m02\_** to be 17 units less than **m01\_** → type: **m01\_x0 - 17**.
  - o Example 2: To constrain the intensity of peak **m03\_** to be half that of **m02\_** → type: **m02\_ampli / 2**



	Label	Model	X0	expression X0	fix X0	Fwhm	expression Fwhm	fix Fwhm	Ampli	expression Ampli	fix Am
✖	m01_	A	Lorentzian	409.725	None	0.001	None		6.431	None	
✖	m02_	E	Lorentzian	392.725	m01_x0 - 17	5.424	None		1038.248	None	
✖	m03_	LO(M)	Lorentzian	385.352	None	10.362	None		519.124	m02_ampli/2	
✖	m04_	TO(M)	Lorentzian	414.523	None	6.666	None		2822.882	None	
✖	m05_	ZO(M)	Lorentzian	463.381	None	26.637	None		296.755	None	
✖	m06_	2LAM	Lorentzian	364.862	None	26.493	None		167.265	None	

### 5.3.3 FitModelControl Panel

Once the desired fit model has been defined (as shown in the **PeakTable** panel), click the button **Fit** to start the fitting process (Figure 10).

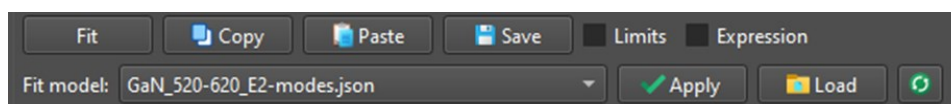


Figure 10: Interface of FitModelControl

If the result is unsatisfactory, adjust the parameters (via **PeakTable** or **SpectraViewer**) and click **Fit** again.

**Copy / Paste Fit Models:** The entire fit model (including spectral range, baseline, and peaks) can be copied and pasted between spectra:

- Select the fitted spectrum and click **Copy**.
- Select one or multiple target spectra and click **Paste**.

**Saving Fit Models:** Users can save the fit model for later use by clicking **Save**. It is recommended to store all models in a predefined folder, which can be specified in the **Settings Panel** (Figure 2). Saved models can then be quickly accessed using the "Fit Model" dropdown menu.

To using a pre-define fit model :

- Select the fit model via the drop-down list. Then click to the **Apply** button to perform the fitting for selected spectra.
- Use can also load a fit model store anywhere in the PC via **Load** button.

### 5.4 Collect and Save Fit Results

SPECTROview provides simple and efficient access to fitted data. To collect best-fit results from all fitted spectra or maps. To save bestfit data:

- Switch to the **Fi Results** tab.
- Click to the Collect button **Collect fit results**.
- Results are then aggregated into a table and displayed in the GUI (Figure 11).

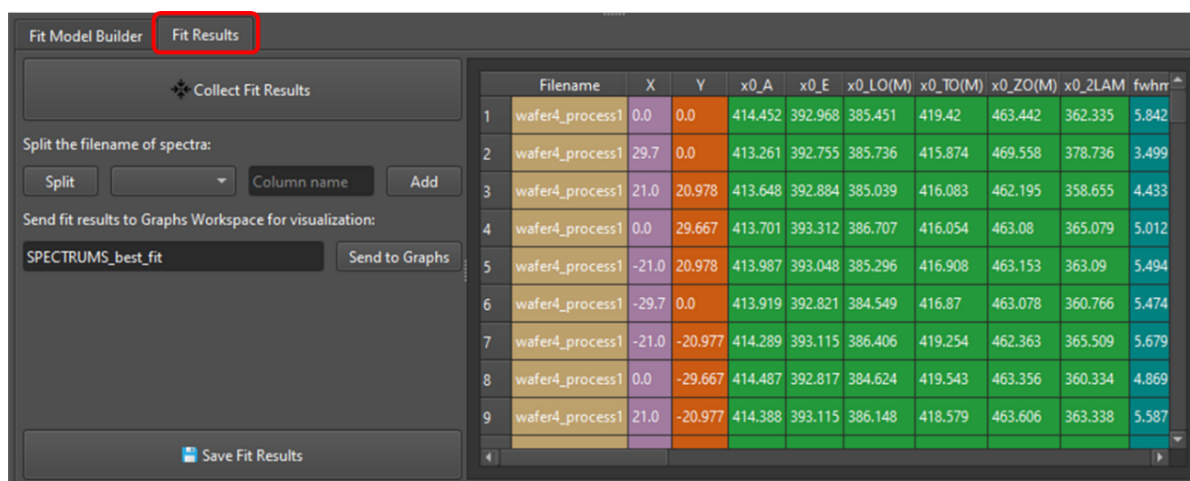
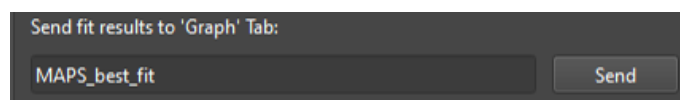



Figure 11: Collect fit results panel

From here, the user can :

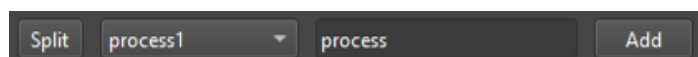
- Name the bestfit dataset and click “**send**” it to the **Graphs** workspace for plotting (cf. Section 6) :



- Or, export the results as an Excel file .

**Split feature:** allows extracting information from filenames and adds it as new columns in the results table.

- Recommendation: Name files wisely using underscores to separate information/parameters (e.g., **Sample1\_ProcessA\_Temp25**).
- Example: In the figure below, the process type (process1, process2) is extracted from file name of each spectra; and added as a new column named "Process"..



## 5.5 Save workspace.

To save the current active workspace, click the **Save** button (2) in the toolbar.



Depending on the active workspace, the file will be saved with a different extension :

(**Maps** workspace → .maps; **Spectra** workspace → .spectra; **Graphs** workspace → .graphs).

## 6. « Graphs » workspace

The **Graphs Workspace** is dedicated to data visualization, with an emphasis on simplicity and speed. The graphical user interface (GUI) of this workspace is shown in Figure 12:

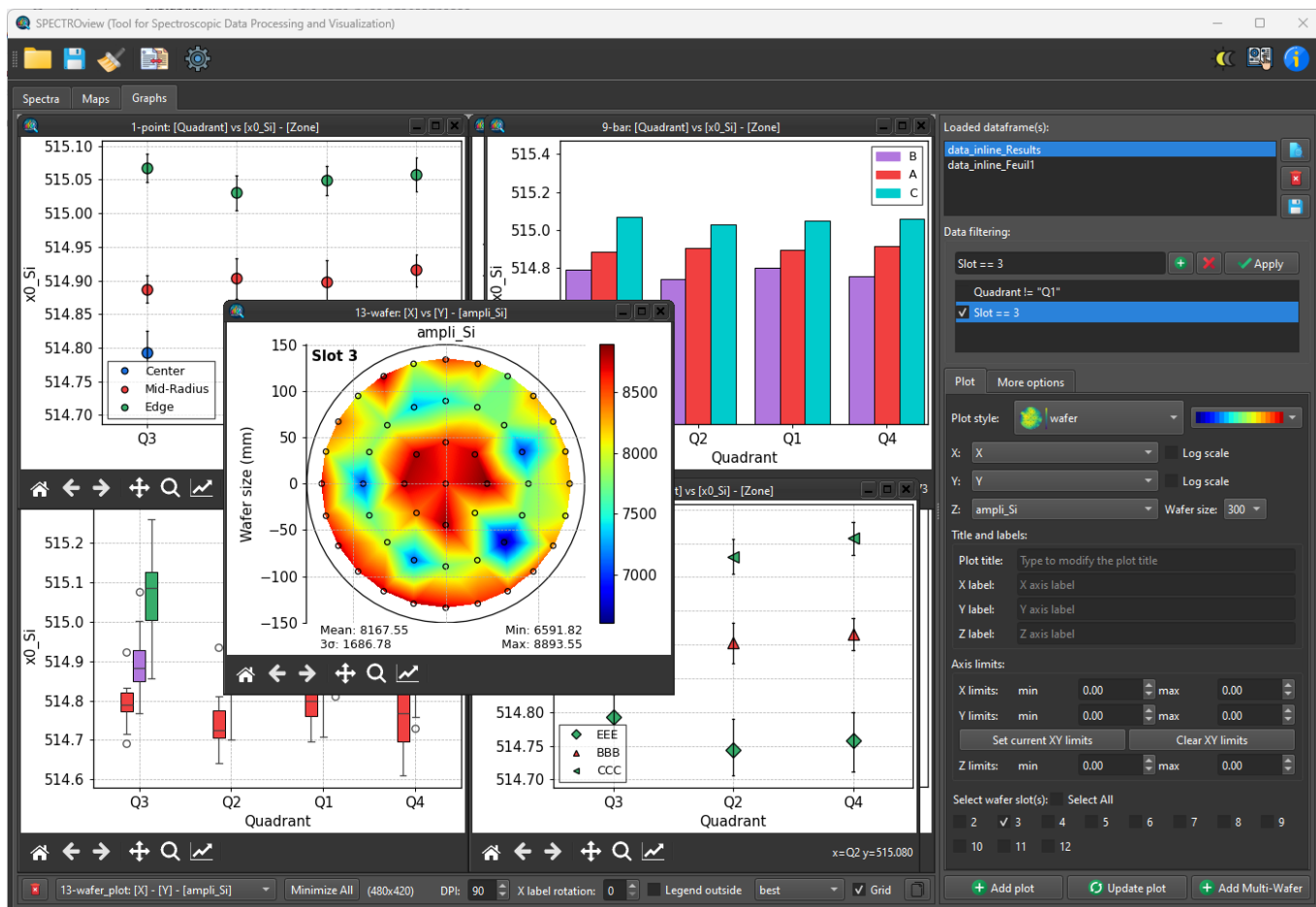


Figure 12: Graphs Workspace interface. The Control Panel is located on the right side, while the Graph Viewer is displayed on the left side.

## 6.1 Loading data

Datasets (dataframes) can be sent directly from other workspaces (Spectra or Maps) or loaded from Excel files. Excel files may contain one or multiple worksheets; each worksheet is loaded into the Graphs Workspace as an independent dataset. Multiple datasets can be loaded simultaneously.

All loaded datasets are displayed in a list widget (cf. top section of the right control panel within Figure 12), from which the user can select the dataset to work with.

## 6.2 Add a new plot

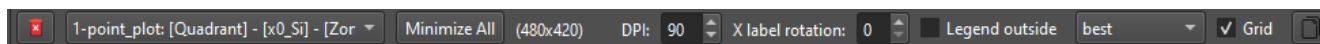
- Selected datasheet among the loaded dataset via the Listbox → X, Y, Z dropdown menu will be populated with all columns available in the selected dataset.
- Select the plot styles : scatter, point, bar, box, line, 2Dmap, wafer
- Specify the label of the x and y axis, as well as the title of the plot (optional).
- Set the limits of the x and y axis (optional)
- Define the label as well as the z limit range (for the heatmap type only) (optional).
- For wafer plot type: → define the wafer size: 100, 200, 300 (mm).
- Click to the Add plot button

## 6.3 Update/Modify existing plot

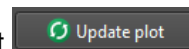
To update/modify or adjusting the any properties an existing plot:



- Select to the plot to be updated/modified.
- Make change(s)/modification(s) via the **ControlPanel** on the right side, such as: plot type, axis labels, axis limits, text size (DPI), axis label rotations, legend box location, grid...



- Click **Update** button to apply the changes/modifications to the selected plot
- Click to the **Copy** button to copy the select figure into Clipboard.



## 6.4 Data filtering feature

The **pandas.DataFrame.query()** method allows you to filter rows from a DataFrame based on a boolean expression. It's a powerful and flexible way to subset your DataFrame to include only the rows that meet specific conditions.

In **SPECTROview**, User can work with the entire dataset or a part of the dataset by using **filtering feature** (Figure 13). The selected dataset can be filtered by typing as following syntax:

**(column\_name) (operator) (value):**

- **(column\_name)**: is the exact header of the column containing the data to be filtered. When the column header contain 'space', you need to enclose them in single or double quotes (see example below).
- **(operator)**: it could be comparison operators ( e.g., **==**, **<**, **>**, **<=**, **>=**, **!=**) and logical operators (e.g., **and**, **or**, **not**) to build complex and multiples conditions.
- **(value)**: it could be numeric or string values. String value must be enclosed in double quotes (cf. example below)

Examples of using Filter features: Assuming that the datasheet containing several columns, including the cloumns whose headers are : **Confocal**, **Thickness**, **a3\_LOM**, **Laser Power** and other columns....

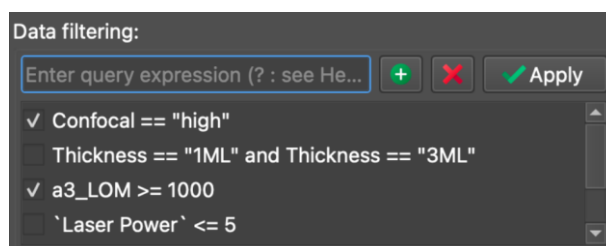


Figure 13: Example of the use of filters for dataframe within SPECTROview

Confocal <b>!=</b> "high"	→ selected all values within column "Confocal" that is different to "high"
Thickness <b>==</b> "1ML" <b>or</b> Thickness <b>==</b> "3ML"	→ selected all values within column "Thickness" that is equal to 1ML <b>OR</b> 3ML
a3_LOM <b>&gt;=</b> 1000	→ selected all values within column "a3_LOM" that is superior to 1000
`Laser Power` <b>&lt;=</b> 5	→ selected all values within column "Laser Power" that is superior or equal to 5.

## 6.5 Other plot customization

Additional view and customization options are available in the **MoreOptions** tab (Figure 14).

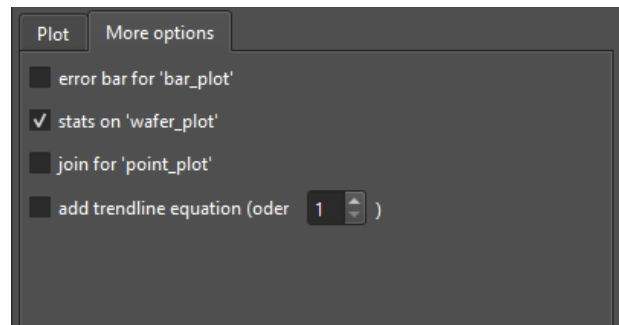


Figure 14: MoreOptions Panel

Users can also customize the legend box (labels, colors, and/or markers) for each curve directly from the plot by clicking on the legend box (Figure 15).

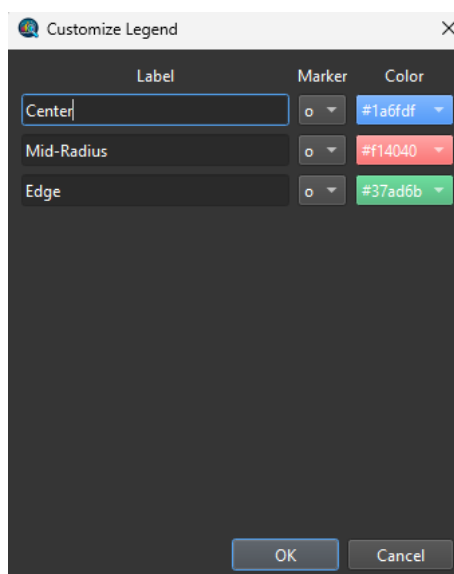


Figure 15: Customize Legend panel displayed when clicking on the plot legend.