

# ETAS Data Operator V1.5



User Guide

## **Copyright**

The data in this document may not be altered or amended without special notification from ETAS GmbH. ETAS GmbH undertakes no further obligation in relation to this document. The software described in it can only be used if the customer is in possession of a general license agreement or single license. Using and copying is only allowed in concurrence with the specifications stipulated in the contract.

Under no circumstances may any part of this document be copied, reproduced, transmitted, stored in a retrieval system or translated into another language without the express written permission of ETAS GmbH.

**© Copyright 2026** ETAS GmbH, Stuttgart

The names and designations used in this document are trademarks or brands belonging to the respective owners.

Data Operator V1.5 | User Guide R01 EN | 04.2026

## Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Intended use	5
1.2	Target group	5
1.3	Classification of Safety Messages	5
1.4	Safety Advice	6
1.5	Data protection	7
1.6	Data and information security	7
1.6.1	Data and storage locations	7
1.6.1.1	License management	7
1.6.2	Technical and organizational measures	7
1.7	Licensing	8
1.7.1	License Variants	8
<b>2</b>	<b>Installation</b>	<b>9</b>
2.1	System requirements	9
2.2	Installing	10
2.3	Uninstalling	10
<b>3</b>	<b>Quick start guide</b>	<b>11</b>
<b>4</b>	<b>Working with Data Operator</b>	<b>12</b>
4.1	Add files	12
4.2	Add a folder	13
4.3	Add operations to the job	14
4.3.1	Add calculated signals	14
4.3.2	Add FMU-based signals	14
4.3.3	Add GPS-based signals	15
4.3.4	Append files	15
4.3.5	Combine files	16
4.3.6	Execute Python script	17
4.3.7	Extract signals	17
4.3.8	Map signal names	18
4.3.9	Merge files	18
4.3.10	Resample signals	19
4.3.11	Shift signals by time	19
4.3.12	Slice files by condition	19
4.3.13	Slice files by duration	20
4.3.14	Trim files	20

4.4	Start the job .....	20
4.5	Edit meta data .....	21
<b>5</b>	<b>CLI guide .....</b>	<b>23</b>
5.1	Working with the CLI .....	23
5.2	Configuring task concurrency .....	24
5.3	Commands .....	25
<b>6</b>	<b>Use cases .....</b>	<b>26</b>
6.1	Creating a LAB file for filtering signals .....	26
6.2	Validating MF4 files .....	26
6.3	Working with signals with device names .....	27
6.4	Exporting and importing a configuration .....	28
6.5	Searching for events using rule-based slicing .....	28
<b>7</b>	<b>User interface .....</b>	<b>30</b>
7.1	Main window .....	30
7.2	Create job .....	31
7.2.1	Sources .....	32
7.2.2	Operations .....	34
7.2.2.1	Add calculated signals .....	35
7.2.2.2	Add FMU-based signals .....	36
7.2.2.3	Add GPS-based signals .....	37
7.2.2.4	Append files .....	38
7.2.2.5	Combine files .....	39
7.2.2.6	Execute Python script .....	40
7.2.2.7	Extract signals .....	41
7.2.2.8	Map signal names .....	41
7.2.2.9	Merge files .....	43
7.2.2.10	Resample signals .....	43
7.2.2.11	Shift signals by time .....	44
7.2.2.12	Slice files by condition .....	44
7.2.2.13	Slice files by duration .....	46
7.2.2.14	Trim files .....	46
7.3	Destination .....	47
7.4	Jobs progress .....	48
7.5	Edit meta data .....	49
<b>8</b>	<b>Contact information .....</b>	<b>51</b>

# 1 Introduction

## 1.1 Intended use

ETAS Data Operator is designed to streamline and standardize the pre-processing of measurement data across different formats, domains, and environments.

Measurement data is often recorded in various formats and may exist in inconsistent or unknown states. In practice, preprocessing is frequently performed using multiple tools and individual workflows, which can lead to non-reproducible and inconsistent results, especially when working with large numbers of files.

ETAS Data Operator addresses this challenge by providing a unified and configurable workflow for data preprocessing.

The software allows users to define and execute customizable processing pipelines that prepare measurement data for further analysis and downstream applications. All processing steps are configured in advance and executed consistently, ensuring reproducible results independent of the user or environment.

The tool supports both interactive usage via a graphical user interface and automated execution via command line or API, enabling integration into local or cloud-based workflows.

ETAS Data Operator is particularly suited for scenarios where large volumes of measurement data must be processed in a consistent, efficient, and scalable manner, such as in automotive development and validation workflows.

## 1.2 Target group

This product is intended for engineers and technical users working with measurement data, particularly in automotive development and validation contexts.

Typical users include development and calibration engineers, as well as data analysts and validation engineers who are responsible for preparing measurement data for further processing and analysis.

The document assumes a basic understanding of measurement data and signal-based data structures, as well as general data processing workflows and standard software usage on Windows systems.

## 1.3 Classification of Safety Messages

Safety messages warn of dangers that can lead to personal injury or damage to property:

**DANGER**

**DANGER** indicates a hazardous situation that, if not avoided, will result in death or serious injury.

**WARNING**

**WARNING** indicates a hazardous situation that, if not avoided, could result in death or serious injury.

**CAUTION**

**CAUTION** indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.

***NOTICE***

**NOTICE** indicates a situation that, if not avoided, could result in damage to physical property.

***ATTENTION***

**ATTENTION** indicates a situation that, if not avoided, could result in damage to digital property like data loss, data corruption and system vulnerability.

## 1.4 Safety Advice

Adhere to the ETAS Safety Advice for Data Operator V1.5, which is available within the Data Operator product. ETAS GmbH cannot be made liable for damage that is caused by incorrect use and not adhering to the safety instructions. Take all information on environmental conditions into consideration before setup and operation (see the documentation of your computer, hardware, etc.).

## 1.5 Data protection

If the product contains functions that process personal data, legal requirements of data protection and data privacy laws shall be complied with by the customer. As the data controller, the customer usually designs subsequent processing. Therefore, he must check if the protective measures are sufficient.

## 1.6 Data and information security

To securely handle data in the context of this product, see the next sections about data and storage locations as well as technical and organizational measures.

### 1.6.1 Data and storage locations

The following sections give information about data and their respective storage locations for various use cases.

#### 1.6.1.1 License management

When using the ETAS License Manager in combination with user-based licenses that are managed on the FNP license server within the customer's network, the following data are stored for license management purposes:

##### Data

- Communication data: IP address
- User data: Windows user ID

##### Storage location

- FNP license server log files on the customer network

When using the ETAS License Manager in combination with host-based licenses that are provided as FNE machine-based licenses, the following data are stored for license management purposes:

##### Data

- Activation data: Activation ID
  - Used only for license activation, but not continuously during license usage

##### Storage location

- FNE trusted storage  
C:\ProgramData\ETAS\FlexNet\fne\license\ts

### 1.6.2 Technical and organizational measures

We recommend that your IT department takes appropriate technical and organizational measures, such as classic theft protection and access protection to hardware and software.

## 1.7 Licensing

A valid license is required to use the software. You can obtain a license in one of the following ways:

- from your tool coordinator
- via the self-service portal on the ETAS website at [www.etas.com/support/licensing](http://www.etas.com/support/licensing)
- via the ETAS License Manager

To activate the license, you must enter the Activation ID that you received from ETAS during the ordering process.

For more information about ETAS license management, see the [ETAS License Management FAQ](#) or the ETAS License Manager help.

To open the ETAS License Manager help

The ETAS License Manager is available on your computer after the installation of any ETAS software.

1. From the Windows Start menu, select **E > ETAS > ETAS License Manager**.  
The ETAS License Manager opens.
2. Click in the ETAS License Manager window and press F1.  
The ETAS License Manager help opens.

### 1.7.1 License Variants

Each license type enables a specific part of the Data Operator workflow. Data Operator separates configuration from execution, which is reflected in the license model. Each license has its own purchase number and must be ordered separately.

#### **DATAOPERATOR\_GUI**

Allows you to start and use the Data Operator graphic user interface (GUI).

The GUI is used to define processing workflows, configure jobs, and trigger their execution. Configurations are created in the GUI and can be reused for automated execution via command line, scripts, or API.

#### **DATAOPERATOR\_WORKER**

Required to execute data processing jobs.

The worker performs the processing of measurement data based on a previously defined configuration. Jobs can be triggered from the GUI as well as via command line, scripts, or API. Each running job requires one worker license, and multiple worker licenses enable parallel execution of jobs.

## 2 Installation

### [2.1 System requirements](#)

### [2.2 Installing](#)

### [2.3 Uninstalling](#)

## 2.1 System requirements

The following minimum system prerequisites have to be met:

Required hardware	1,0 GHz PC 4 GB RAM Graphics with a resolution of at least 1024 x 768, 32 MB RAM
Required operating system	Windows® 10, Windows® 11
Required free disk space	1 GB (not including the size for application data)

The following system prerequisites are recommended:

Required hardware	2,0 GHz Dual-Core PC or equivalent 32 GB RAM Graphics with a resolution of 1680 x 1050, 128 MB RAM
Required operating system	Windows® 10, Windows® 11
Required free disk space	>2,0 GB



### Note

The number of parallel processing tasks used by Data Operator can be controlled via the environment variable **DATAOP\_CONCURRENT\_TASKS\_LIMIT**. This option can be useful to reduce memory usage on low-resource systems or to optimize performance on high-performance systems. For instructions on how to apply the setting, refer to ["Configuring task concurrency" on page 24](#).

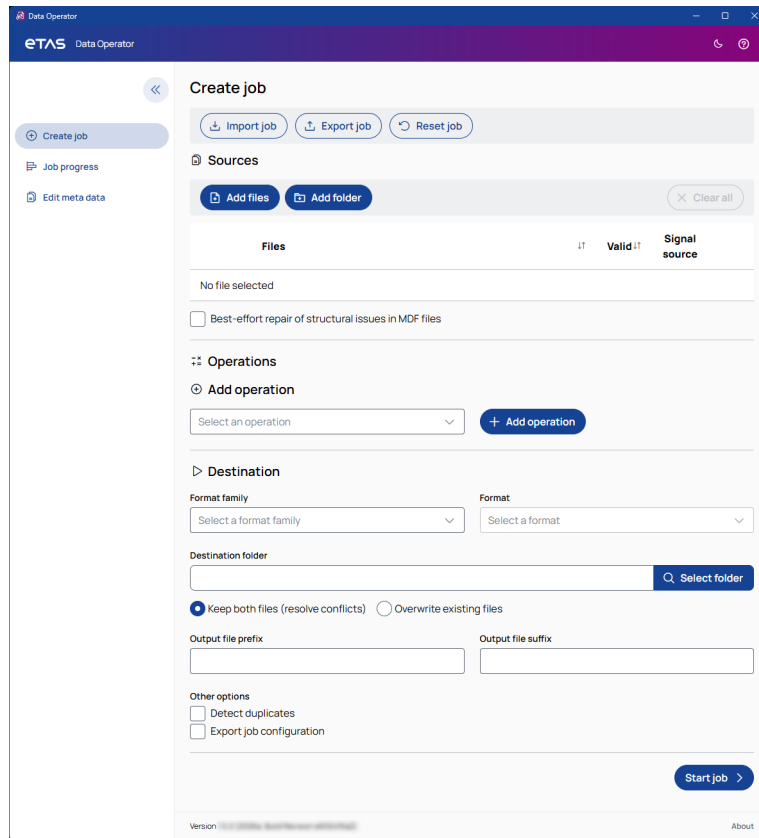
## 2.2 Installing

1. Navigate to the location of the Data Operator X.X x64.exe and double-click it.  
The installation wizard opens.
  2. Select the **Destination Folder** for the Data Operator installation.  
Click **Browse** to browse for the destination folder.
  3. Click **Install**.
  4. After the installation is finished, select whether to directly run Data Operator by activating/deactivating the **Run Data Operator** checkbox.
  5. Click **Finish**.
- ⇒ The installation is completed. You can now use Data Operator:

## 2.3 Uninstalling

1. Navigate to the Data Operator installation folder.
  2. Double-click Uninstall DataOperatorX.X.exe.  
The uninstall wizard opens.
  3. Click **Uninstall**.
  4. After the uninstallation is finished, click **Finish**.
- ⇒ The uninstallation is completed.

## 3 Quick start guide



### Start a job

1. In the **Sources** step of the **Create job** tab, [add files](#) or [add a folder](#) to the job.
  2. (Optional) "[Add operations to the job](#)" on page 14 you want to perform to the measurement files
  3. In the **Destination** step, select the [destination format and folder](#) and click **Start job**.
- ⇒ The job starts. You can track its progress in the [Jobs progress](#) tab.

## 4 Working with Data Operator

This chapter describes the individual steps and operations available in the **Create job** view, where you define how your measurement data should be processed—from input selection to output configuration.

To import a previously saved configuration and restore all settings and operations, use the **Import configuration** button. See also, "[Export a configuration](#)" on page 28.

### [4.1 Add files](#)




### [4.2 Add a folder](#)

### [4.3 Add operations to the job](#)

### [4.4 Start the job](#)


### [4.5 Edit meta data](#)

### 4.1 Add files






1. In the **Sources** step, click  .
2. In the file selection dialog, select the measurement files you want to add to the job.  
The validity of MF4 files is automatically checked upon adding. The **Valid** column displays a green check mark for valid files and a red cross for corrupt ones.  
If a corrupt MF4 file is detected, you can enable the **Best-effort repair of structural issues in MDF files** checkbox to attempt a best-effort repair. Unreadable sections may be removed during the repair, so it is recommended to work on a copy of the original file.
3. Click **Open**.  
The selected files are added to the job.
4. Use the **Signal source** checkbox to enable or disable individual files for signal selection.  
By default, the first five added files are activated as signal sources.
5. To remove a file, click  .
6. To remove all added files and folders, click  .

### **BLF files**

If BLF files are added, a signal definition file is required to interpret the CAN data.


1. To select a signal definition file, click .
  2. In the file selection dialog, select a DBC or ARXML file.
  3. If a DBC file is selected, enter an integer value in the **CAN bus ID** input field.
  4. If an ARXML file is selected:
    - i. In the **CAN bus ID** input field, enter an integer value.
    - ii. From the **Bus name** drop-down list, select a bus.
- ⇒ The configuration is applied to all added BLF files.

## 4.2 Add a folder


1. In the **Sources** step, click .
2. In the folder selection dialog, select the folder containing your measurement files and click **Select Folder**.  
The folder appears as an expandable entry in the sources list.
3. Use the **Signal source** checkbox to enable the folder as a source for signal selection in operations.  
By default, no signals from folders are used as signal sources. You can enable them manually.
4. To validate all MF4 files in the folder, click .  
If corrupt MF4 files are found during validation, you can enable the **Best-effort repair of structural issues in MDF files** checkbox to attempt a best-effort repair. Unreadable sections may be removed during the repair, so it is recommended to work on copies of the original files.
5. To filter files inside the folder, expand it and use the **File selection filter** field. Wildcards (\*, ?) or regular expressions can be used.
6. To refresh folder contents, click .  
If any files were modified externally, deselect and reselect them to reload their signals.
7. To remove the folder from the job, click .
8. To clear all added files and folders, click .

### **BLF files**

If the selected folder contains BLF files, a signal definition file must be provided.

1. To select a signal definition file, click .
  2. In the file selection dialog, select a DBC or ARXML file.
  3. If a DBC file is selected, enter an integer value in the **CAN bus ID** input field.
  4. If an ARXML file is selected:
    - i. In the **CAN bus ID** input field, enter an integer value.
    - ii. From the **Bus name** drop-down list, select a bus.
- ⇒ The configuration is applied to all added BLF files.

### 4.3 Add operations to the job

1. In the **Add operation** step, select an operation from the drop-down list.
  2. Click .
- ⇒ The selected operation is added to the job.

The following operations can be added to the job:

#### 4.3.1 Add calculated signals

Add a new signal by defining a formula based on existing signal data.

1. In the **Output signal name** input field of the **Add calculated signal** step, enter a name for the new signal to be created.
2. In the **Formula** input field, enter a formula using existing signal names:
  - The formula uses Python syntax. NumPy is available.
  - Only a subset of Python features is supported. For details, see ["Add calculated signals" on page 35](#).

**Example:** `np.sqrt(signal_name)`

#### Note


The result of the formula must have the same length as the input signals.

Modifying time values or sampling is not supported within formulas.

3. To define another calculated signal, click .

#### 4.3.2 Add FMU-based signals

Add new signals based on a Functional Mock-up Unit (FMU) model.

1. After adding the **Add FMU-based signals** operation, click .
2. In the file selection dialog, select the FMU file you want to use.
3. In the **Time steps** input field, enter a positive value and select the required unit from the drop-down list.
4. In the **Input mapping** section, for each entry in the **FMU input** column, select the corresponding signal from the **Signal name** drop-down list.  
The available signals in the **Signal name** drop-down list are based on the selected signal sources. See, "[Signal source](#)" on page 32.
5. In the **Output mapping** section, for each entry in the **FMU output** column, enter a name in the **Signal name** field.

### Note

FMU-based signals may replace existing signals with the same name.  
Changing the FMU file resets all input and output mappings.

### 4.3.3 Add GPS-based signals

GPS data can be annotated with information retrieved from © OpenStreetMap.

1. In the **Longitude signal name** input field, enter the exact name of your longitude signal.
2. In the **Latitude signal name** input field, enter the exact name of your latitude signal.
3. Under **Signals to add**, select the annotations you want to add to your data.

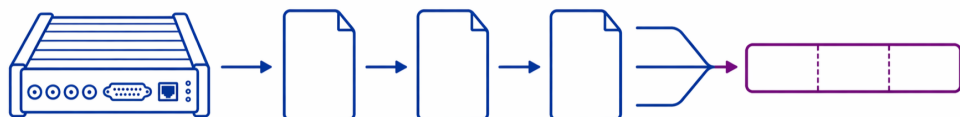
See also, "[Add GPS-based signals](#)" on page 37.

### Note

The GPS-based annotation feature retrieves data from © OpenStreetMap.  
If your system uses a proxy, ensure that the environment variables **HTTP\_PROXY** and **HTTPS\_PROXY** are set correctly. Otherwise, the operation may fail during job execution.

**Example error:** "Failed to resolve 'overpass-api.de'".

### 4.3.4 Append files



Use this operation to combine multiple measurement files into a single output file by appending them sequentially.

Use this operation when a recorder produces multiple consecutive measurement files that need to be processed as one continuous dataset.

1. After adding the **Append files** operation, select how the files are ordered via the **Measurement order** drop-down list:
  - **Alphanumerical file name order (A→Z)**: Files are sorted in ascending order based on their file names.
  - **Alphanumerical file name order (Z→A)**: Files are sorted in descending order based on their file names.
  - **Measurement date time order**: Files are sorted based on their measurement timestamps.
  - **Sources order**: Files are processed in the order in which they were added in the "Sources" on page 32 step.
2. (Optional) In the **Fixed gap** input field, enter a time value to insert a gap between each appended file.
3. To select the unit, use the drop-down list.



#### Note

Using a fixed gap may affect time-based analyses if the inserted gap does not reflect actual measurement conditions. Verify the resulting timeline before further processing.

### 4.3.5 Combine files

The **Combine files** operation will be deprecated in a future release. Use the **Append files** or **Merge files** operations instead, which provide improved configuration options and enhanced processing behavior.

You can combine multiple files into a single output file.

There are two modes available:

#### Merge

Merge files based on their timestamps and align them on a common timeline.

1. After adding the **Combine files** operation, select **Merge**.
2. Select whether to **Keep** or to **Reduce** the gap between data.
3. If you select **Reduce**, you must define a **Fixed gap** and a **Gap threshold**. To select the unit, use the drop-down list.
  - **Gap threshold**: Gaps equal to or larger than this value will be reduced.
  - **Fixed gap**: Specifies the new gap size to apply to gaps that meet or exceed the threshold.

### Append

Append files one after another into a single file.

1. After adding the **Combine files** operation, select **Append**.
2. Select the order in which files are combined:
  - **Creation time**: Based on the creation time of each file.
  - **Sequence**: Based on the order files were added in the **Source** step.
3. In the **Fixed gap** input field, enter the time gap to insert between files. To select the unit, use the drop-down list.

### 4.3.6 Execute Python script

Use this operation to execute a custom Python script on the measurement data.

1. In the **Python script path** input field of the **Execute Python script** operation, select or enter the path to your Python script file.
2. (Optional) Activate the **Execute individually per measurement file** checkbox to control how the script is executed:
  - **Checked**: The script is triggered once per measurement file, allowing parallel execution. This is faster but cannot access data from other measurement files.
  - **Unchecked**: The script is triggered once for all files together, which allows operations combining data across multiple files. This mode is slower but supports data aggregation scenarios.



#### Note


Refer to the **Python scripting guide** for detailed instructions on writing custom operations.

Make sure that all script dependencies are installed and accessible in the Python environment used by Data Operator.

### 4.3.7 Extract signals

Filtering signals requires a LAB file that lists the signals you want to keep.

To create such a file using Data Operator, see ["Creating a LAB file for filtering signals" on page 26](#).

1. After adding the **Extract signals** operation, click .
2. In the file selection dialog, select the LAB file you want to use for filtering.
3. Click **Open**.


### 4.3.8 Map signal names


Create a new signal using values from a list of candidate signals, selected in order of priority.


The original signals remain in the output file. The mapped signals are added in addition to the existing ones.

1. In the **Mapped signal name** input field of the **Map signal names** operation, enter a name for the new signal.
2. In the **Candidate signal** section, define one or more source signals:
  - i. In the **Signal name** input field, enter the name of the original signal to map from.
  - ii. (Optional) In the **Device** input field, enter the name of the device the signal belongs to.

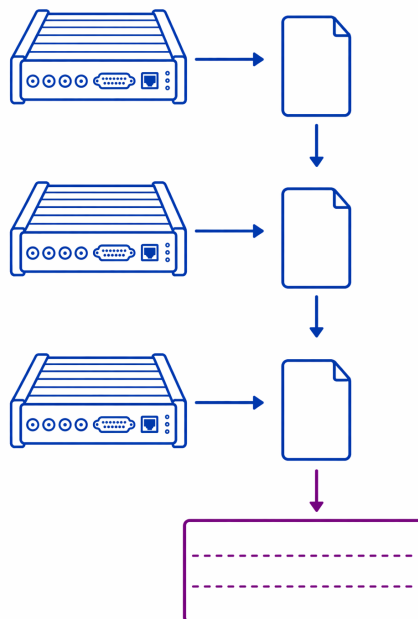
The first matching signal from the list is used.

3. Click  to add additional signals to the priority list. Signals are evaluated from top to bottom.

4. To remove a candidate signal entry, click .

5. To define additional mapped signals, each with its own candidate list, click .

### 4.3.9 Merge files



Use this operation to combine multiple measurement files into a single output file by merging them.

This operation is intended for use cases where multiple recorders capture different signals at the same time.

1. After adding the **Merge files** operation, select how time offsets between measurement files are handled using the **Time offset strategy** drop-down list:
  - **None**: All measurement files are combined using their unmodified time signals. No additional alignment is applied, and existing time axes remain unchanged.
  - **Align based upon measurement date time**: Measurement files are aligned using the recorded timestamps stored in their metadata. Any differences in start time result in corresponding time offsets, so that signals are synchronized according to their actual recording time.

#### 4.3.10 Resample signals

You can define a new sampling rate, which can effectively reduce the file size.

1. After adding the **Resample Signals** operation, enter a positive value in the **Resampling period** input field.
2. Select the time unit from the drop-down list.

#### 4.3.11 Shift signals by time

Apply a time offset to specific signals to align them with other data.

1. In the **Signal list** input field of the **Shift signals by time** operation, enter the names of the signals to shift.  
Separate multiple signal names with semicolons (;).
2. In the **Time offset** input field, enter the time value to shift the signals.  
Use a positive value to delay the signals or a negative value to shift them earlier.
3. To select the unit, use the drop-down list.

#### 4.3.12 Slice files by condition

Files can be divided into rule-based slices with user-defined conditions. This lets you isolate specific events or time ranges in measurement data.

1. After adding the **Slice files by condition** operation, select a **Slice mode**: **Condition**, **One sided trigger**, or **Two sided trigger**.
2. Enter the condition or conditions using Python syntax. NumPy is available. Only a subset of Python features is supported. For details, see "[Slice files by condition](#)" on page 44.  
Multiple signals can be combined.

- **Condition** mode keeps only the data points that satisfy the rule.
- **One sided trigger** creates slices that start when the condition first becomes true, extend for the configured **Duration**, and are shifted by the **Offset**. Overlapping slices are merged into a single slice.  
Use this mode to inspect time spans around an event, such as a threshold being exceeded.
- **Two sided trigger** creates slices between a **From** condition and an **Until** condition. Offsets are applied to both slice boundaries.  
Use this mode to inspect time spans between two events.

### 4.3.13 Slice files by duration

Files can be divided into time-based slices of a defined duration.

1. After [adding the Slice files by duration operation](#), enter a positive value in the **Slice duration** input field to define the length of each slice.
2. To select the unit, use the drop-down list.


### 4.3.14 Trim files


Files can be trimmed at a single time step by using only one input field, or at two time steps by filling in both fields. If a field is left empty, the original start or end of the file is used by default.

1. After [adding the Trim files operation](#), enter a positive value in the **Start** and/or **End** input field.  
The file is cut after the specified time step relative to the start time of the measurement file.
2. To select the unit, use the drop-down list.

## 4.4 Start the job

When you are satisfied with all the settings made in the **Operations** steps, you can start the job.





1. In the **Destination** step, select a **Format family** from the drop-down list.
2. Select a **Format** the corresponding drop-down list.  
The available formats depend on the format family.
3. To select the destination folder, click . The button is blue with a magnifying glass icon and the text 'Select folder'.
4. In the folder selection dialog, select the destination folder.
5. Click **Select Folder**.
6. Select an option to handle naming conflicts in the destination folder:

- **Keep both files (resolve conflicts)**: Keeps all files by appending a numeric suffix (e.g., \_1, \_2, etc.) to the new file name. The number increases automatically to avoid overwriting existing files.
  - **Overwrite existing files**: Overwrites existing files with the same name.
7. (Optional) In the **Output file prefix** and/or **suffix** input fields, enter custom text to add to the output file names.
  8. (Optional) Activate any of the following checkboxes:
    - **Detect duplicates**: Detects duplicate signals in measurement files. The results of this operation can be found in the job log.
    - **Export job configuration**: Saves the configuration with all settings as JSON file to the destination folder. It can be used with the [Data Operator CLI](#) or imported back into Data Operator, see "[Import a configuration](#)" on page 28.
  9. Click  .
- ⇒ The job starts. You can track its progress in the [Jobs progress](#) tab.

## 4.5 Edit meta data

Use the **Edit meta data** page to view, modify, and add meta data for multiple measurement files at once.

### Edit meta data

1. In the navigation sidebar, click **Edit meta data**.
  2. Click  .
  3. In the file selection dialog, select the MF4 measurement files whose meta data you want to edit and click **Open**.  
The selected files are listed in the table. Each row represents one measurement file.
  4. To add a new meta data field, click  .  
A new column is added to the table.
  5. Edit the values directly in the table cells.
  6. To remove all added files and attributes from the table, click  .
  7. To save the edited meta data, click  .
- ⇒ The original measurement files are saved with the edited meta data strings.

 **Note**

Use forward slashes (/) in attribute names to define hierarchical meta data nodes.

**Example:**

Driver Info/Surname creates a root node Driver Info with a subnode Surname.

If an attribute name is already used for a root-level attribute that carries data, you cannot create a hierarchical meta data node with the same name.

If an attribute name is already used as a hierarchical meta data node with child nodes, you cannot create a root-level attribute with the same name.

A meta data node that has child nodes cannot carry data itself.

Modify a column

1. Click the column header to select the column.
2. Right-click to open the context menu.

The following actions are available:

- **Insert column** – adds a column to the left of the selected column
- **Delete column**
- **Rename column**

Select and edit multiple cells

1. Select one or more cells using standard mouse and keyboard interactions:
  - Click and drag with the mouse.
  - Click while holding SHIFT or CTRL.
  - Press SHIFT together with the arrow keys.
2. To clear the selected cells, press DELETE.
3. To copy, cut, or paste cell values, use standard keyboard shortcuts:
  - CTRL+C to copy
  - CTRL+X to cut
  - CTRL+V to paste

## 5 CLI guide

The Command Line Interface (CLI) allows automated data processing with Data Operator.

It is especially useful for incorporating tasks such as automatic filtering and resampling of measurement data into CI/CD pipelines or other automation workflows.



### Note

A valid **DATAOPERATOR\_WORKER** license is required during processing when using the CLI functionality of Data Operator.

### 5.1 Working with the CLI

The commands available in the Data Operator CLI are described in the section ["Commands" on page 25](#).

To use the CLI, you must first create a configuration using the Data Operator GUI application.

#### *Generate a workflow*

1. In Data Operator make your desired settings as described in ["Working with Data Operator" on page 12](#).
  2. In the **Destination** step, activate the **Export configuration** checkbox.
  3. Click **Start job**.
- ⇒ The configuration is saved as JSON file to the destination folder.

#### *Open Data Operator with PowerShell*

1. Open the Data Operator installation folder.
  2. Click in the address bar at the top of the file explorer.
  3. Type `powershell` and press ENTER.  
PowerShell opens in the current directory.
  4. To start Data Operator CLI, enter:  
`.\DataOperatorCLI.exe`
- ⇒ You can now use Data Operator CLI.

#### *Open Data Operator with Command Prompt (cmd)*

1. Open the Data Operator installation folder.
2. Click in the address bar at the top of the file explorer.
3. Type `cmd` and press ENTER.

Cmd opens in the current directory.

- To start Data Operator CLI, enter:

```
DataOperatorCLI.exe
```

⇒ You can now use Data Operator CLI.

### *Run commands with PowerShell*

```
.\DataOperatorCLI.exe run [command]
```

#### **EXAMPLE**

```
.\DataOperatorCLI.exe run --help
```

### *Run commands with cmd*

```
DataOperatorCLI.exe run [command]
```

#### **EXAMPLE**

```
DataOperatorCLI.exe run --help
```

## 5.2 Configuring task concurrency

Data Operator executes certain operations in parallel to reduce processing time.

By default, up to **three concurrent tasks** are executed.

The number of concurrent tasks can be configured by setting the environment variable **DATAOP\_CONCURRENT\_TASKS\_LIMIT**.

Adjusting this value allows you to balance performance and memory usage:

- **Lower values** reduce memory consumption and help avoid out-of-memory errors.
- **Higher values** can improve processing speed on systems with sufficient CPU performance and available RAM.

If the environment variable is not set, Data Operator uses the default limit of three concurrent tasks.

### *Setting the environment variable with Command Prompt (cmd)*

- Open the **Start** menu.
- Type cmd and press ENTER.
- In the **Command Prompt** window, enter the following command:

```
setx DATAOP_CONCURRENT_TASKS_LIMIT <value>
```

Replace **<value>** with the desired number of concurrent tasks.

4. Press ENTER to execute the command.
5. Close the **Command Prompt** window.
6. Restart any open **Command Prompt** or **Data Operator** sessions so that the new environment variable is applied.

**EXAMPLE**

To limit processing to a single concurrent task:

```
setx DATAOP_CONCURRENT_TASKS_LIMIT 1
```

To **verify the setting** in a new Command Prompt, execute: `echo %DATAOP_CONCURRENT_TASKS_LIMIT%`

## 5.3 Commands

`-h/--help`

Displays all available commands along with their descriptions.

`-v/--version`

Outputs the current version number of Data Operator.

`-d/--destination <path>`

Sets the output path where the processed data will be saved.

`-s/--source <path>`

Specifies one or more source paths to the files or folders to be processed.

`-w/--workflow <path>`

Specifies the path to the configuration file (\*.json) you want to use.

 **Note**

If a path contains spaces, enclose it in double quotation marks.

**EXAMPLE**

```
DataOperatorCLI.exe run -w C:\your\path\example.json -s
"C:\your\source\file or folder" -d C:\y-
our\destination\path
```

## 6 Use cases

### [6.1 Creating a LAB file for filtering signals](#)

### [6.2 Validating MF4 files](#)

### [6.3 Working with signals with device names](#)



### [6.4 Exporting and importing a configuration](#)

### [6.5 Searching for events using rule-based slicing](#)

## 6.1 Creating a LAB file for filtering signals


If you want to filter signals from a measurement file, you need a LAB file containing only the signals you want to keep (see "Extract signals" on page 17).

To create such a LAB file with Data Operator, follow the steps below.


1. In the **Sources** step of the **Create job** tab, [add the measurement files](#) whose signals you want to filter later.
  2. In the **Destination** step, select **LabFile** from the **Format family** drop-down list.
  3. From the **Format** drop-down list, select any LAB file format.
  4. Select the destination folder with  .
  5. Click  .  
The LAB file is saved to the selected folder.
  6. In the Jobs progress tab, click Open destination folder for the respective job.
  7. Open the LAB file in a text editor of your choice (e.g., Notepad++).
  8. Delete the signals you want to remove from the measurement file later.
  9. Save the LAB file.
- ⇒ You can now use the created LAB file to filter the signals from your measurement files using the [Filter signals](#) operation.

## 6.2 Validating MF4 files

Check the validity of single MF4 files


1. In the **Sources** step, click  .
  2. In the file selection dialog, choose the MF4 files you want to check.
- ⇒ Each file is automatically validated upon upload. The result is shown in the **Valid** column. A green check mark indicates a valid file, while a red cross indicates a corrupt one.

### Check the validity of MF4 files of a whole folder

1. In the **Sources** step, click .
2. In the folder selection dialog, choose the folder containing the MF4 files you want to check.



#### **Note**

If a filter is applied, only the visible (filtered) files in the folder are validated.

3. To validate all files in the folder, click .
- ⇒ The files are validated. The results are shown in the **Valid** column. A green check mark indicates a valid file, while a red cross indicates a corrupt one.

## 6.3 Working with signals with device names

Signals that include a device name are not supported in most processing steps. To work with such signals, you must first create a mapped signal without the device reference using the **Signal Mapping** step.

1. In the **Sources** step of the **Create job** tab, add the measurement file that contains the signals with device names.
2. In the **Add operation** step, select **Signal mapping** from the drop-down list and click .
3. In the **Signal mapping** step, go to the **Mapped signal name** input field and enter a name for the new signal.
4. In the **Candidate signal** section:
  - i. In the **Signal name** input field, enter the name of the original signal.
  - ii. In the **Device** input field, enter the name of the device the signal belongs to.
5. (Optional) To add another mapping, click  and repeat from step 2.
6. In the **Destination** step, select the [destination format and folder](#), then click **Start job**.
 

⇒ The file containing the mapped signals without device names is saved to the destination folder. [Add this file](#) to a new job to use the signals in subsequent processing steps.

## 6.4 Exporting and importing a configuration

You can export a job configuration to reuse the same processing setup later or import an existing configuration to restore all job settings and operations.

An exported configuration file is also required to work with the [Data Operator CLI](#).

### Export a configuration

1. In the **Sources** step of the **Create job** tab, [add a file](#) to the job.
2. "[Add operations to the job](#)" on page 14 that you want to export to the configuration file.
3. Set the [destination format and folder](#).


4. At the top of the **Create job** tab, click .

Or

5. In the **Destination** step, activate the **Export configuration** checkbox and click **Start job**.

⇒ The configuration file is saved as JSON file in the selected destination folder.


### Import a configuration

1. At the top of the **Create job** tab, click .
  2. In the file selection dialog, select a previously saved configuration file (\*.json).
- ⇒ All operations and settings defined in the configuration are restored.

## 6.5 Searching for events using rule-based slicing

If you want to search for specific events or conditions in your measurement data, you can use the **Rule based slice** operation.

This allows you to define logical expressions that isolate only the relevant parts of your data.

1. In the **Sources** step of the **Create job** tab, [add the measurement files](#) that contain the data you want to search.
2. In the **Add operation** step, select **Rule based slice** from the drop-down list and click .
3. In the **Condition** field, enter a Python expression that evaluates to true for rows that match your desired condition.

**Example:** `vehicleSpeed > 50 and brakePressure > 0`

4. In the **Destination** step, select the [destination format and folder](#), then click **Start job**.

5. Once the job is finished, got to the **Jobs progress** tab.
  6. On the respective job card, click **Open destination folder**.
- ⇒ You can now review only the data segments where the defined condition was met – for example, to analyze specific events such as braking during high-speed driving.

## 7 User interface

### [7.1 Main window](#)

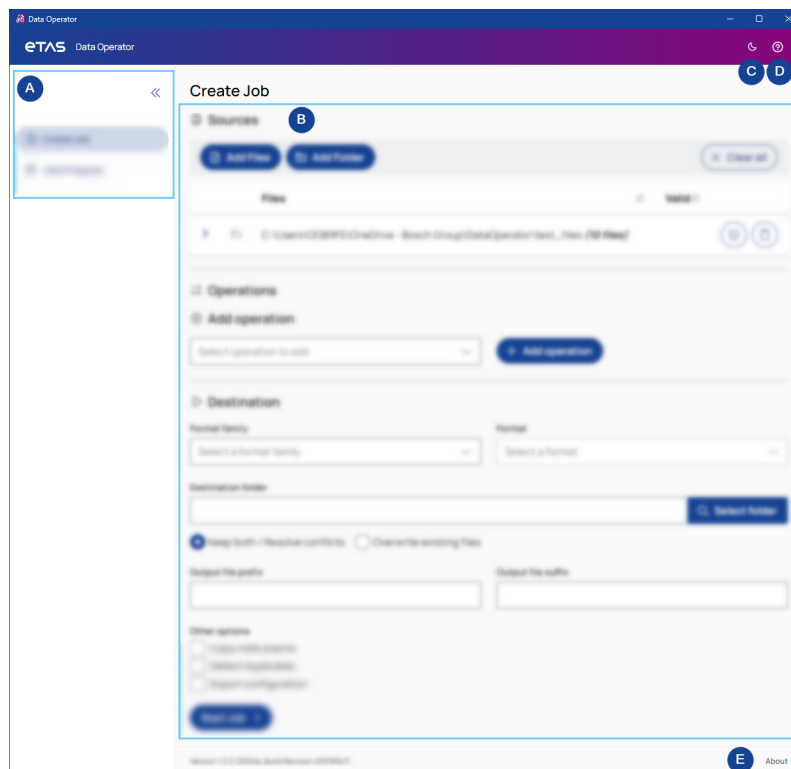
### [7.2 Create job](#)

### [7.3 Destination](#)

### [7.4 Jobs progress](#)

### [7.5 Edit meta data](#)

## 7.1 Main window



### A. **Navigation sidebar**

Use the navigation sidebar to switch between the "Create job" on the next page, "Jobs progress" on page 48, and "Edit meta data" on page 49 pages.

To expand or collapse the sidebar, click the arrows in the top-right corner.

### B. **Content**

Displays the main content, depending on the selected view.

### C. **Dark/Light mode**

To toggle between light and dark mode, click the sun/moon in the top right corner.

### D. **Help**

Opens the PDF User Guide.

#### E. **About**

Shows the meta data of the app.

## 7.2 Create job

The **Create job** view provides a centralized interface for adding sources, selecting operations, and configuring destination settings such as file format and output folder.

 **Import configuration**

Import a previously saved configuration to restore all settings and operations. See also, "Export a configuration" on page 28.

 **Export job**

Export a configuration to save all settings and operations.

 **Reset job**

Resets the current job. All settings are cleared, and all added files and folders are removed.

The following settings can be made to the job:

### [7.2.1 Sources](#)

### [7.2.2 Operations](#)

[7.2.2.1 Add calculated signals](#)

[7.2.2.2 Add FMU-based signals](#)

[7.2.2.3 Add GPS-based signals](#)

[7.2.2.4 Append files](#)

[7.2.2.5 Combine files](#)

[7.2.2.6 Execute Python script](#)

[7.2.2.7 Extract signals](#)

[7.2.2.8 Map signal names](#)

[7.2.2.9 Merge files](#)

[7.2.2.10 Resample signals](#)

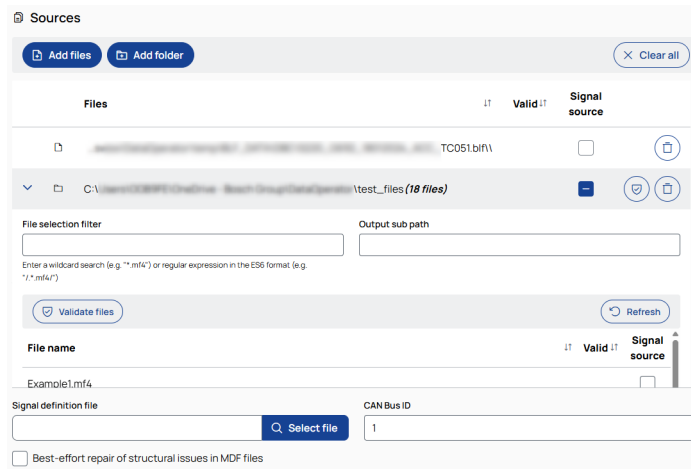
[7.2.2.11 Shift signals by time](#)

[7.2.2.12 Slice files by condition](#)

[7.2.2.13 Slice files by duration](#)

[7.2.2.14 Trim files](#)

## 7.2.1 Sources



### Add files

Opens a file selection dialog to choose files to add to the job.  
See also, "Add files" on page 12.

### Add folder

Opens a folder selection dialog to choose a folder to add to the job.  
See also, "Add a folder" on page 13.

### Clear all

Removes all added files and folders from the job.

## Files

Lists all files in the job, sorted by most recently added.

To sort the list alphabetically, ascending or descending, click the column header.

## Valid

This column shows whether an MF4 file is valid or corrupt.

A green check mark indicates a valid file, while a red cross indicates a corrupt one.

## Signal source

Enables or disables a file or folder as a source for signal selection in operations.

- When you add individual files, the first five files are automatically enabled as signal sources.
- When you add a folder, no files are preselected as signal sources.
- You can manually enable or disable any files or folders as signal sources.

If a file has changed after the signals were loaded, deselect and reselect it to refresh the signal database.



Validates all MF4 files in the folder.


A green check mark indicates a valid file, while a red cross indicates a corrupt one.

Single files are validated during upload.

When a filter is applied, only the visible (filtered) files are validated.

## Signal definition file

Select a signal definition file to interpret CAN data in BLF files.

Click  to open a file selection dialog and choose a DBC or ARXML file.

## CAN bus ID

Enter an integer value that defines the CAN bus used for signal extraction.

## Bus name

Select the CAN bus from the drop-down list.

This field is only available if an ARXML file is selected.

## Best-effort repair of structural issues in MDF files

Scans MDF files for structural issues and attempts a best-effort repair.

Unreadable or damaged sections may be removed to restore file integrity, which can lead to partial data loss.

It is recommended to repair a copy of the original file.

Behavior when executed in a job:

### Repair only

If the **Best-effort repair of structural issues in MDF files** option is enabled and the job is executed without any additional operations, all input MDF files are scanned, files with structural issues are repaired, and the repaired versions are saved to the destination folder.

### Repair before operations

If the **Best-effort repair of structural issues in MDF files** option is enabled and the job contains one or more operations, all input files are repaired first. After this repair step, the configured operations are applied to all resulting (initially valid and repaired) files.

This ensures that all subsequent processing is performed on structurally valid MDF files.



Removes the file from the job.



Refreshes the folder. Updates changes made to the folder outside the application.

## 7.2.2 Operations



### Add operation

To select a processing step (operation) to include in the job, use the drop-down list.



After selecting an operation, click the Add operation button to add it to the job.

The following operations can be added to the job:

[7.2.2.1 Add calculated signals](#)

[7.2.2.2 Add FMU-based signals](#)

[7.2.2.3 Add GPS-based signals](#)

[7.2.2.4 Append files](#)

[7.2.2.5 Combine files](#)

[7.2.2.6 Execute Python script](#)

[7.2.2.7 Extract signals](#)

[7.2.2.8 Map signal names](#)

[7.2.2.9 Merge files](#)

[7.2.2.10 Resample signals](#)

[7.2.2.11 Shift signals by time](#)

[7.2.2.12 Slice files by condition](#)

[7.2.2.13 Slice files by duration](#)

[7.2.2.14 Trim files](#)

### 7.2.2.1 Add calculated signals

Adds a new calculated signal with its own formula and name.

#### Output signal name

Enter a name for the new signal that will be created.

#### Formula

Enter a formula using existing signal names. Only signals without a device name are supported.

The supported autocompletion is based on the signal sources defined in the **Sources** step.

Formulas use Python syntax, but with limited supported features. NumPy is available via `np`.

The result of the formula must have the same length as the input signals.

Modifying time or sampling rate is not supported.

**Example:** `np.sqrt(signal_name)`

#### Note

Only the following Python keywords are supported: **and, elif, else, if, in, is, not, or, return, time**.

Formulas are not validated during editing. Errors only appear during job execution.

Signal names must be valid Python identifiers:

- Start with a letter or underscore.
- Contain only letters, digits, underscores.
- No spaces or special characters (e.g., @, #, !, \, etc., except underscore).
- No Python reserved keywords (e.g., `if`, `else`, `while`, `def`, `true`, `false`, etc.).



Adds a new calculated signal with its own formula definition.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.2 Add FMU-based signals

#### Functional mockup unit (FMU) model

Enter the path to the FMU file or select it using .

After selecting a valid FMU file, the FMU inputs and outputs are automatically detected and displayed.

#### Time steps

Enter a positive value to define the time interval at which the FMU is executed to calculate output signals.

To select the unit, use the drop-down list.

#### Input mapping

**FMU input:** Displays the input variables defined in the FMU model

**Signal name:** Select the measurement signal that is used as input for the corresponding FMU variable. The available signals are based on the selected signal sources. See, "[Signal source](#)" on page 32.

#### Output mapping

**FMU output:** Displays the output variables defined in the FMU model.

**Signal name:** Enter the name of the signal to be created.



#### Note

FMU-based signals may replace existing signals with the same name. Changing the FMU file resets all input and output mappings.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.3 Add GPS-based signals

#### Longitude signal name

Enter the exact name of the longitude signal.

#### Latitude signal name

Enter the exact name of the latitude signal.

#### Signals to add

Select the types of annotations to add to your GPS data. The information used is obtained from © OpenStreetMap.

**Speed limit:** Maximum speed limits for road sections.

**Road type:** The classification of the road segment. The mapping is as follows:

- Motorway: 1
- Trunk: 2
- Primary: 3
- Secondary: 4
- Tertiary: 5
- Unclassified: 6
- Residential: 7
- Service: 8
- Motorway link: 9
- Trunk link: 10
- Primary link: 11
- Secondary link: 12
- Tertiary link: 13

**Distance to traffic light:** Distance from the GPS position to the nearest traffic light.

**GPS velocity:** Instantaneous velocity calculated from changes in GPS position over time.

### Note

The GPS-based annotation feature retrieves data from © OpenStreetMap. If your system uses a proxy, ensure that the environment variables **HTTP\_PROXY** and **HTTPS\_PROXY** are set correctly. Otherwise, the operation may fail during job execution.

**Example error:** "Failed to resolve 'overpass-api.de'".



To remove the operation from the job, click the trash can button.

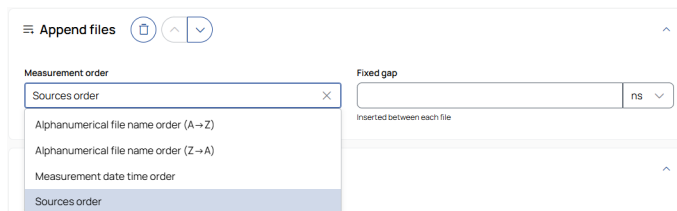


To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

## 7.2.2.4 Append files



### Measurement order

Defines the order in which files are appended:

**Alphanumerical file name order (A→Z):** Sorts files in ascending order based on their file names.

**Alphanumerical file name order (Z→A):** Sorts files in descending order based on their file names.

**Measurement date order:** Sorts files based on their measurement timestamps.

**Sources order:** Uses the order defined in the "Sources" on page 32 step.

### Fixed gap

Defines a fixed time gap between appended files.

To select the unit, use the drop-down list.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.5 Combine files

The **Combine files** operation will be deprecated in a future release. Use the **Append files** or **Merge files** operations instead, which provide improved configuration options and enhanced processing behavior.



#### Combination

##### Append

Appends all files in the job sequentially into a single output file.

##### Order

**Creation time:** Appends files based on the creation time of the measurement files.

**Sequence:** Appends files in the order they were added in the **Source** step.

##### Fixed gap

Inserts a fixed time gap between the appended files. To select the unit, use the drop-down list.

##### Merge

Merges the data from the job's files into a single file based on their timestamps on a shared timeline.

##### Gap

**Keep:** Keeps the original gaps between files unchanged.

**Reduce:** Reduces gaps based on the following settings.

- **Gap threshold:** Gaps equal to or larger than this value will be reduced. To select the unit, use the drop-down list.
- **Fixed gap:** Specifies the new gap size to apply to gaps that meet or exceed the threshold. To select the unit, use the drop-down list.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.6 Execute Python script

Executes a custom Python script to perform user-defined processing on measurement data.

#### Python script path

Enter or browse to the path of the Python script file you want to execute. See the example scripts in the installation folder for reference.

#### Execute individually per measurement file

Defines how the Python script is executed:

- **Enabled** – The script runs once per measurement file, processing them in parallel. This is faster but cannot use data from other files.
- **Disabled** – The script runs once for all files together, allowing operations that combine data from multiple measurement files. This mode is slower.

#### Note

Make sure the Python environment used by Data Operator includes all required dependencies for your script.

Refer to the Python scripting guide for available interfaces and best practices.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.7 Extract signals



Opens a file selection dialog where you can select the LAB file to be used for filtering the signals.

Alternatively, you can type the path in the input field.

To create a LAB file for filtering signals with Data Operator, see "[Creating a LAB file for filtering signals](#)" on page 26.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.8 Map signal names

Creates a new signal using values from a list of candidate signals, selected in order of priority. The new signals are created in the output file but the original signals are not removed from the measurement data.

### Mapped signal name

Enter the name for the new, mapped signal.

### Candidate signals

Provide the original signal names to pull data from. The first available match (from top to bottom) is used.

This can be used as a fallback or priority list for multiple sensors.

**Signal name:** The original name of the signal to match.

**Device:** Optionally specify the device the signal belongs to. Leave empty to apply the mapping across all devices.

### Load config\_signals.csv

You can import a configuration file named `config_signals.csv` to define multiple signal mappings at once.

The file follows the same structure used in ETAS EATB.

Example files can be found in the installation folder.



You can remove any entry using the trash can icon.



Adds another signal to the list of candidates.



Adds a new mapped signal entry, each with its own list of candidate signals.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.9 Merge files



#### Time offset handling

Defines how time offsets between measurement files are applied:

**None:** Aligns all measurement files to a common start time. Each file starts at time zero, regardless of its original timestamp.

**Measurement date time:** Preserves the original time offsets between measurement files. The signals are aligned according to their measurement timestamps.



To remove the operation from the job, click the trash can button.

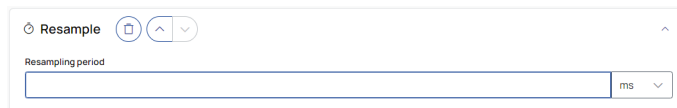


To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.10 Resample signals



#### Resampling period

Enter a positive integer to define the resampling period. To select the unit, use the drop-down list.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.11 Shift signals by time

Applies a time offset to the specified signals to align them with other data.

#### Signals

Type the names of the signals to shift and use ENTER to confirm. The supported autocomplete is based on the signal sources defined in the **Sources** step.

#### Time offset

Enter the amount of time to shift the signals. Use a positive value to delay the signal or a negative value to shift it earlier. To select the unit, use the drop-down list.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.12 Slice files by condition

Slices a measurement file into smaller segments based on a user-defined rule. You can select different slice modes depending on how the slice boundaries should be detected.

#### Slice mode

Select how the slice boundaries are calculated:

- **Condition:** Keeps only the values that meet the condition.
- **One sided trigger:** Creates slices with a defined duration after a condition becomes true. Overlapping slices are merged into one slice.
- **Two sided trigger:** Creates slices between a start condition (**From**) and an end condition (**Until**).

## Condition

Enter a condition using Python syntax, but with limited supported features. NumPy is available.

Only data points that satisfy the condition remain. Multiple signals can be combined in the expression.

**Example:** `vehicleSpeed > 50`

### Note

Only the following Python keywords are supported: **and, elif, else, if, in, is, not, or, return, time**.

Formulas are not validated during editing. Errors only appear during job execution.

Signal names must be valid Python identifiers:

- Start with a letter or underscore.
- Contain only letters, digits, underscores.
- No spaces or special characters (e.g., @, #, !, \, etc., except underscore).
- No Python reserved keywords (e.g., `if`, `else`, `while`, `def`, `true`, `false`, etc.).

## One sided trigger

Creates a slice whenever the condition becomes true.

The slice starts when the condition first becomes true, is extended by the configured **Duration**, and is shifted by the **Offset**. To select the unit, use the drop-down list.

Overlapping slices are automatically merged into a single slice.

**Example:** A rule such as `vehicleSpeed > 50` creates a slice from the time the speed first exceeds 50, extended by the configured duration and shifted by the offset.

## Two sided trigger

Creates a slice between two independent conditions.

The **From** condition defines when the slice starts; the **Until** condition defines when the slice ends. Offsets are applied to both slice boundaries. To select the unit, use the drop-down list.

Each valid pair of conditions results in a slice.

**Example:** For `vehicleSpeed < 50` until `vehicleSpeed > 100`, a slice is created for the time span where the vehicle is between these thresholds.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.13 Slice files by duration

#### Slice duration

Enter a positive integer to define the duration of each data slice. To select the unit, use the drop-down list.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

### 7.2.2.14 Trim files

#### Note

Files can be trimmed at a single time step by using only one input field, or at two time steps by filling in both fields. If a field is left empty, the original start or end of the file is used by default.

## Start

Enter a positive value to define the start point of the trimmed file. To select the unit, use the drop-down list.

## End

Enter a positive value to define the end point of the trimmed file. To select the unit, use the drop-down list.



To remove the operation from the job, click the trash can button.



To move the operation up or down in the job sequence, use the arrow buttons next to the trash can icon.



To expand or collapse the operation, click the arrow in the top-right corner of the operation panel.

## 7.3

### Destination

The screenshot shows the 'Destination' configuration panel. It includes a 'Format family' dropdown menu set to 'MDF V4.x', a 'Format' dropdown menu set to 'MDF 4.11', a 'Destination folder' input field with a 'Select folder' button, radio buttons for 'Keep both / Resolve conflicts' (selected) and 'Overwrite existing files', 'Output file prefix' and 'Output file suffix' input fields, and a section for 'Other options' with checkboxes for 'Copy md4 events', 'Detect duplicates', and 'Export configuration'. A 'Start Job' button is at the bottom.

### Format family

To select the destination file format family, use the drop-down list.

### Format

To select the destination file format based on the selected format family, use the drop-down list.

### Destination folder



Opens a folder selection dialog where you can choose the destination folder to save the destination file to. Alternatively, you can type the path in the input field.

Select an option to handle naming conflicts in the destination folder:

**Keep both files (resolve conflicts):** Keeps all files by appending a numeric suffix (e.g., \_1, \_2, etc.) to the new file name. The number increases automatically to avoid overwriting existing files.

**Overwrite existing files:** Overwrites existing files with the same name.

## Output file prefix/suffix

In the input fields, enter a prefix/suffix you want to add to the file names.

## Other options

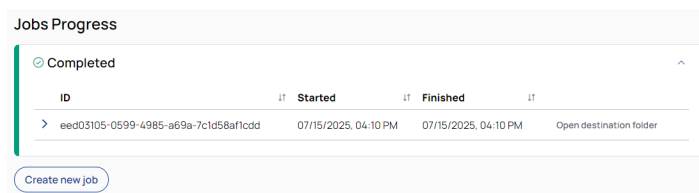
**Detect duplicates:** Detects duplicate signals in measurement files. The results of this operation can be found in the job log.

**Export job configuration:** Saves the configuration with all settings as JSON file to the destination folder. It can be used with the [Data Operator CLI](#) or imported back into Data Operator, see "[Import a configuration](#)" on page 28.

**Start job** >

Starts the job with all the settings you made and opens the **Jobs progress** step, where you can see the status of the job's progress.

## 7.4 Jobs progress



ID	Started	Finished	
eed03105-0599-4985-a69a-7c1d58af1cdd	07/15/2025, 04:10 PM	07/15/2025, 04:10 PM	Open destination folder

Create new job

### Status panel

Shows the status and meta data of the jobs.

To expand or collapse the panel, click the arrow in the top-right corner.

### Status

**Running:** The progress bar shows the progress status.

**Completed:** The **Open destination folder** button opens the destination folder for the job files.

**Failed:** The job failed.

### ID

Displays the ID of the job.

To sort the list alphabetically, ascending or descending, click the column header.

To expand, click the arrow next to the ID:

**Completed:** The path of the destination folder is displayed. Click to open the folder.

**Failed:** The error message is displayed. To open the error log file, click

Show log

## Started/Finished

Displays the time and date when the job was started and when it was finished.

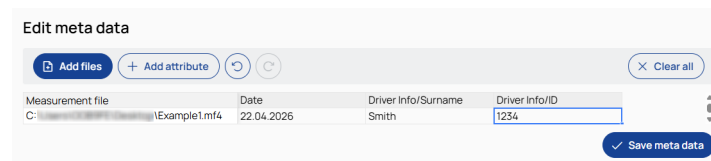
To sort the list alphabetically, ascending or descending, click the column header.

Create new job

Jumps to the **Sources** step where you can start a new job.

## 7.5

### Edit meta data



The **Edit meta data page** provides a centralized interface to view, modify, and add meta data of multiple measurement files in a table.

Add files

Opens a file selection dialog to choose measurement files whose meta data you want to edit.

+ Add attribute

Adds a new meta data field as a column to the table.

Use forward slashes (/) in attribute names to define hierarchical meta data nodes.

A meta data node that has child nodes cannot carry data itself.

If a root-level attribute with data already exists, you cannot create a hierarchical meta data node with the same name.

If a hierarchical meta data node with child nodes already exists, you cannot create a root-level attribute with the same name.



## Undo/redo

**Undo** reverts the last change made in the table.

**Redo** restores the last change that was previously undone.

✕ Clear all

Removes all selected files and all added attributes from the table.

## Table

Displays one measurement file per row.

The first column shows the measurement file, and additional columns represent attributes.

Columns can be selected via the column header. A context menu provides options to insert, delete, or rename columns.

Cells can be edited directly. Multiple cells can be selected using standard mouse and keyboard interactions. Standard keyboard shortcuts are supported.

✓ Save meta data

Writes the edited meta data to the selected measurement files.

## 8 Contact information

### Technical Support

For details of your local sales office as well as your local technical support team and product hotlines, take a look at the ETAS website:

[www.etas.com/hotlines](http://www.etas.com/hotlines)

ETAS offers trainings for its products:

[www.etas.com/academy](http://www.etas.com/academy)



### ETAS Headquarters

ETAS GmbH

Borsigstraße 24	Phone:	+49 711 3423-0
70469 Stuttgart	Fax:	+49 711 3423-2106
Germany	Internet:	<a href="http://www.etas.com">www.etas.com</a>