



ibaAnalyzer

Interface for File Extraction

Manual Part 5
Issue 8.1

Manufacturer

iba AG
Koenigswarterstrasse 44
90762 Fuerth
Germany

Contacts

Main office +49 911 97282-0
Fax +49 911 97282-33
Support +49 911 97282-14
Engineering +49 911 97282-13
E-mail iba@iba-ag.com
Web www.iba-ag.com

Unless explicitly stated to the contrary, it is not permitted to pass on or copy this document, nor to make use of its contents or disclose its contents. Infringements are liable for compensation.

© iba AG 2023, All rights reserved.

The content of this publication has been checked for compliance with the described hardware and software. Nevertheless, discrepancies cannot be ruled out, and we do not provide guarantee for complete conformity. However, the information furnished in this publication is updated regularly. Required corrections are contained in the following regulations or can be downloaded on the Internet.

The current version is available for download on our web site www.iba-ag.com.

Version	Date	Revision	Author	Version SW
8.1	08-2023	GUI new, interactively license-free	ts, rm, mm	8.1.0

Windows® is a brand and registered trademark of Microsoft Corporation. Other product and company names mentioned in this manual can be labels or registered trademarks of the corresponding owners.

Contents

1	About this documentation	4
1.1	Target group.....	4
1.2	Notations	4
1.3	Used symbols.....	5
1.4	Documentation structure	6
2	Introduction	7
2.1	File extraction with ibaAnalyzer	7
2.2	Functions and application.....	7
2.3	Installation	7
3	Data Extractor	8
3.1	Extractor output	10
3.2	Archive profile assignment	11
3.3	Info columns	15
3.3.1	Modify info columns.....	17
3.4	Computed columns.....	18
3.5	Diagnosis and notification	20
3.5.1	Diagnostic log	20
3.5.2	Notifications	21
3.6	Renumbering	22
4	Supported file formats	24
4.1	iba format (.dat file).....	24
4.2	Text file (CSV)	25
4.3	COMTRADE	28
4.4	TDMS	28
4.5	Apache Parquet	29
4.6	Matlab	32
5	Extraction via the command line	34
6	Automatic extraction using ibaDatCoordinator.....	35
7	Support and contact.....	37

1 About this documentation

This documentation describes the function and application of the software *ibaAnalyzer*.

1.1 Target group

This manual addresses in particular the qualified professionals who are familiar with handling electrical and electronic modules as well as communication and measurement technology. A person is regarded as professional if he/she is capable of assessing safety and recognizing possible consequences and risks on the basis of his/her specialist training, knowledge and experience and knowledge of the standard regulations.

This documentation addresses in particular professionals who are in charge of analyzing measured data and process data. Because the data is supplied by other iba products the following knowledge is required or at least helpful when working with *ibaAnalyzer*:

- Operating system Windows
- *ibaPDA* (creation and structure of the measuring data files)

1.2 Notations

In this manual, the following notations are used:

Action	Notation
Menu command	Menu <i>Logic diagram</i>
Calling the menu command	<i>Step 1 – Step 2 – Step 3 – Step x</i> Example: Select the menu <i>Logic diagram – Add – New function block</i> .
Keys	<Key name> Example: <Alt>; <F1>
Press the keys simultaneously	<Key name> + <Key name> Example: <Alt> + <Ctrl>
Buttons	<Key name> Example: <OK>; <Cancel>
Filenames, paths	<i>Filename, Path</i> Example: <i>Test.docx</i>

1.3 Used symbols

If safety instructions or other notes are used in this manual, they mean:

Danger!



The non-observance of this safety information may result in an imminent risk of death or severe injury:

- Observe the specified measures.
-

Warning!



The non-observance of this safety information may result in a potential risk of death or severe injury!

- Observe the specified measures.
-

Caution!



The non-observance of this safety information may result in a potential risk of injury or material damage!

- Observe the specified measures
-

Note



A note specifies special requirements or actions to be observed.

Tip



Tip or example as a helpful note or insider tip to make the work a little bit easier.

Other documentation



Reference to additional documentation or further reading.

1.4 Documentation structure

This documentation describes the functionality of the *ibaAnalyzer* software in detail. It is designed both as a tutorial as well as a reference document.

In addition to this documentation, you can examine the version history in the main menu, *Help – Version history* (file [versions.htm](#)) for the latest information about the installed version of the program. This file not only lists the bugs that have been eliminated but also refers to extensions of the system in note form.

In addition, special "NewFeatures..." documentation comes with any software update that includes significant new features, which provides a more detailed description of the new features.

The state of the software to which the respective part of this documentation refers is listed in the revision table on page 2.

The *ibaAnalyzer* documentation (PDF and printed version) is divided into six separate parts. Each part has its own section and page numbering beginning at 1, and is updated independently.

Part	Title	Content
Part 1	Introduction and Installation	General notes, licenses and add-ons Installation and program start User interface
Part 2	Working with <i>ibaAnalyzer</i>	Working with data file and analysis, presentation features, macro configuration, filter design, preferences, printing, export, interfaces to <i>ibaHD-Server</i> , <i>ibaCapture</i> and report generator
Part 3	Expression builder	Directory of all calculation functions in the expression builder, including explanation
Part 4	Database interface	Working with data from databases, connecting to the database, writing iba measurement data to databases, extracting the data from the database and analyzing the data.
Part 5	Interface for file extraction	Functions and settings for extracting data from iba data files to external file formats
Part 6	Application examples	<i>In preparation</i>

2 Introduction

2.1 File extraction with ibaAnalyzer

The file extraction function in *ibaAnalyzer* is a standard function that you can use interactively without an additional license. The function allows you to extract measurement data acquired in the iba data file format into standard formats, which in turn can be imported by other programs.

Supported formats are the iba format (*.dat), text files (CSV or ASCII), Apache Parquet, Matlab (*.mat), COMTRADE and TDMS. For further details, see chapter ↗ *Supported file formats*, page 24.

Notes for users of already existing solutions with ibaAnalyzer-File-Extract and USB Marx dongle

Even after updating *ibaAnalyzer* to v8.x and *ibaDatCoordinator* v3.x, you can continue to use USB Marx dongles with the licenses *ibaAnalyzer-DB* or *ibaAnalyzer-File-Extract* as usual.

2.2 Functions and application

The file extraction is an integral function of *ibaAnalyzer* and is activated during the installation of *ibaAnalyzer*. You can easily extract, transform and load all data in iba data file format (*.dat) originating from *ibaPDA*, *ibaQDR*, *ibaLogic*, *ibaScope* or third-party applications using the *ibaFiles* library.

Summary of the major functions

- Loading measurements from iba data file
- Transformation of data or generation of new virtual signals
- Extracting data to various formats
- Automated processing with *ibaDatCoordinator*

Other documentation



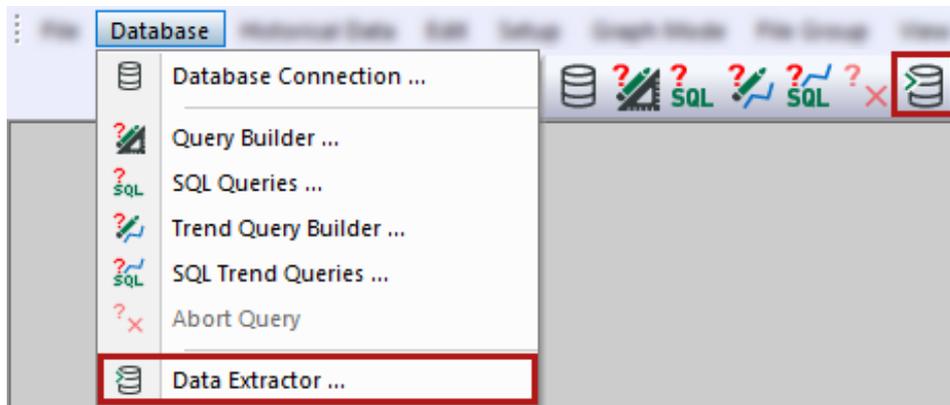
For the usage of *ibaDatCoordinator*, see the corresponding manual or ↗ *Automatic extraction using ibaDatCoordinator*, page 35.

2.3 Installation

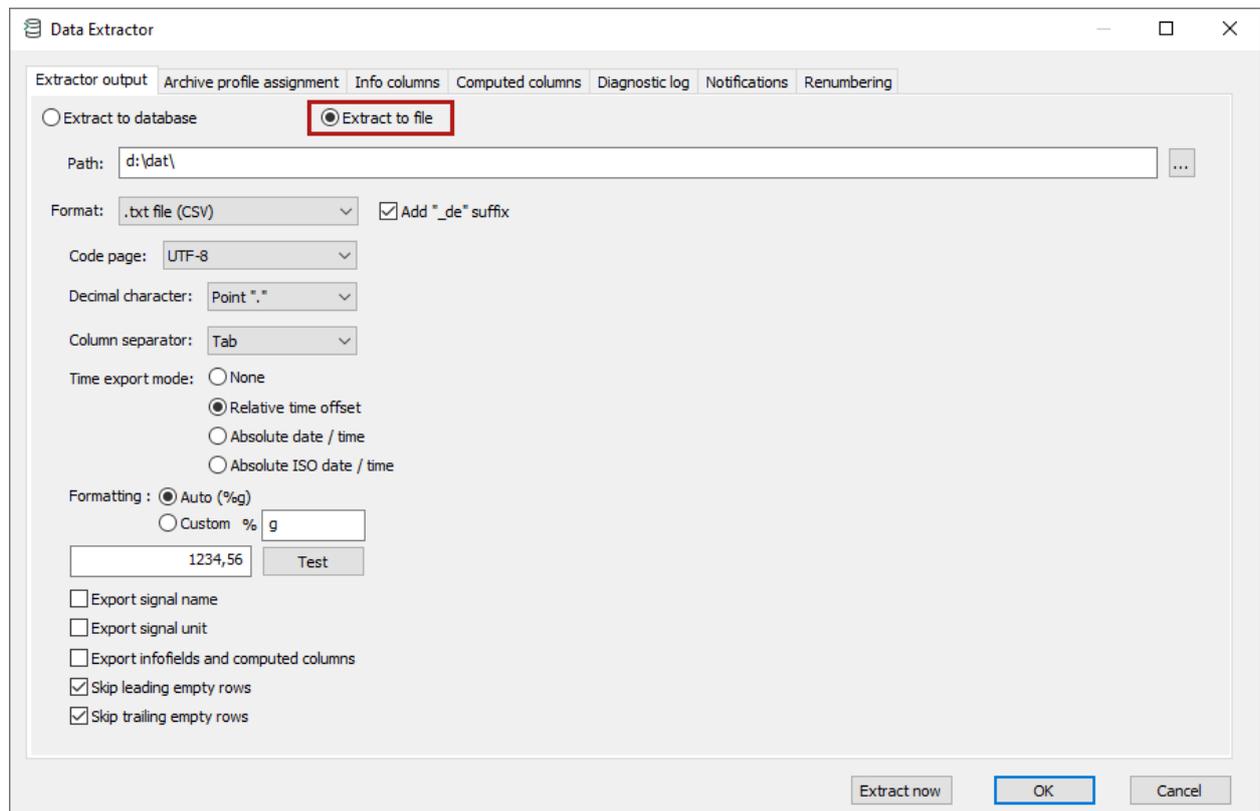
The file extraction function is automatically installed and activated together with *ibaAnalyzer*. During installation, no special steps are required.

3 Data Extractor

Use the dialog *Data Extractor* to configure the file extract. You can open the Data extractor either via the menu *Database – Data Extractor* or via the corresponding button on the toolbar.



The *Extractor Output* tab appears automatically, for further details, see [↗ Extractor output](#), page 10.



Settings for data extraction to file

On the *Extractor Output* tab, select the option *Extract to file*.

→ Thereby all registers adjust the respective settings to the data extraction into a file.

The following chapters describe the individual tabs of the data extractor with the *Extract to file* option selected.

Buttons on each tab

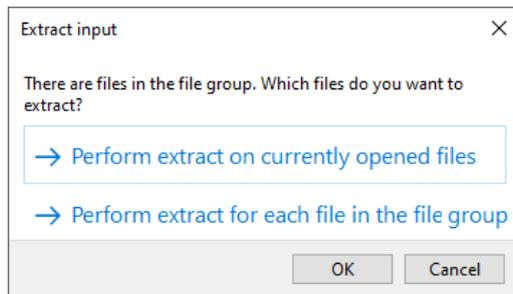
All tabs have the following buttons:

- <Extract now>
ibaAnalyzer creates a file based on the current settings for file extraction. By default, all extracted filenames consist of the source filename augmented by "_de". If extraction from the same source file is repeated, the extracted files are indexed starting with "_00".
- <OK>
ibaAnalyzer saves all current settings withing the dialog and closes the window.
- <Cancel>
ibaAnalyzer discards all changed settings and closes the window.

Tip



If *ibaAnalyzer* opens multiple files in the file group (e.g. from a conditional search using *ibaHD-Server*), you can apply the extraction to all files contained in the file group. When you click the <Extract Now> button, a corresponding dialog appears.

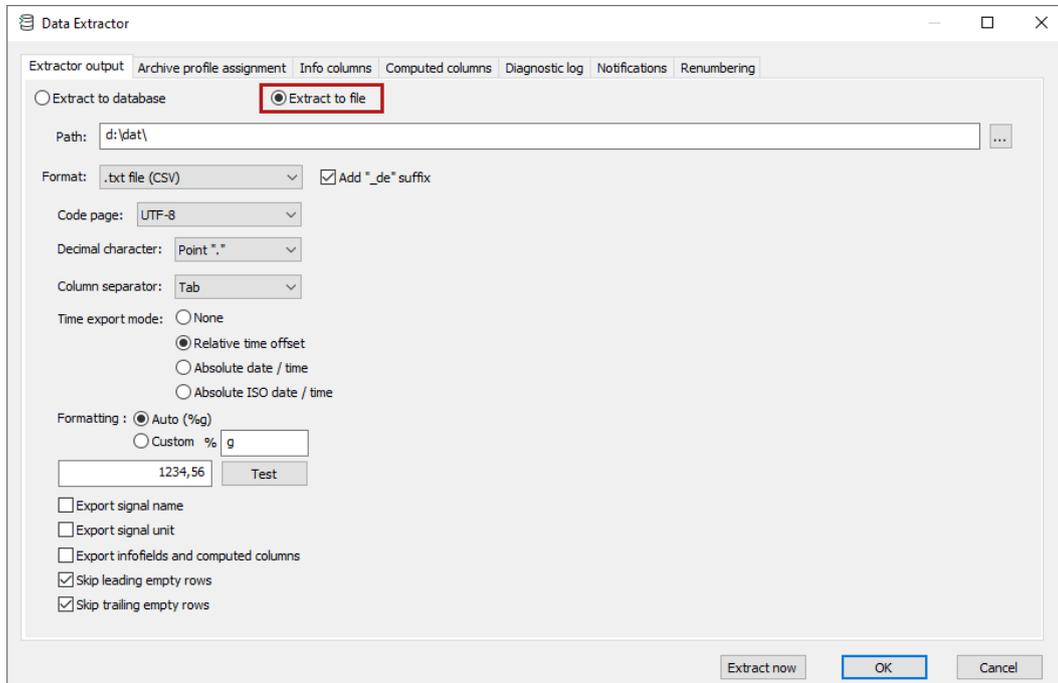


To perform the extraction for multiple files in the file group (2nd option), you need the *ibaDatCoordinator-File-Extract* license.

3.1 Extractor output

The *Extractor Parameters* tab automatically opens first when you open the data extractor.

Select the *Extract to file* option. Here you can specify the location and format of the extracted file.



The Extract to database option extracts data into a database, see also part 4, *Database Interface*.

ibaAnalyzer automatically saves the extracted files in the specified path. If the directory does not exist, it is created automatically. Use the <...> button to browse the file system if necessary.

Filename

File extraction creates output files with the same name as the input file and an additional "_de" suffix. If you disable the *Add "_de" suffix* option, the original filename is used. The filename extension depends on the desired output format.

Use this tab to specify the output file format. For further information, see chapter [Supported file formats](#), page 24.

Note



If you call the analysis with files for an extraction from *ibaDatCoordinator*, you have to define the location of the extracted files there. This overwrites the location defined in *ibaAnalyzer*. In case the directory is password protected, you have to enter the username and password in *ibaDatCoordinator*.

3.2 Archive profile assignment

In the *Archive profile assignment* tab, you can configure archiving profiles that define the sampling rate (time-based or length-based). You then assign the profile to the signals for extraction. You can configure more than one profile. You can assign only 1 profile to a signal.

To create an archiving profile, at least one data file must be open for extraction.

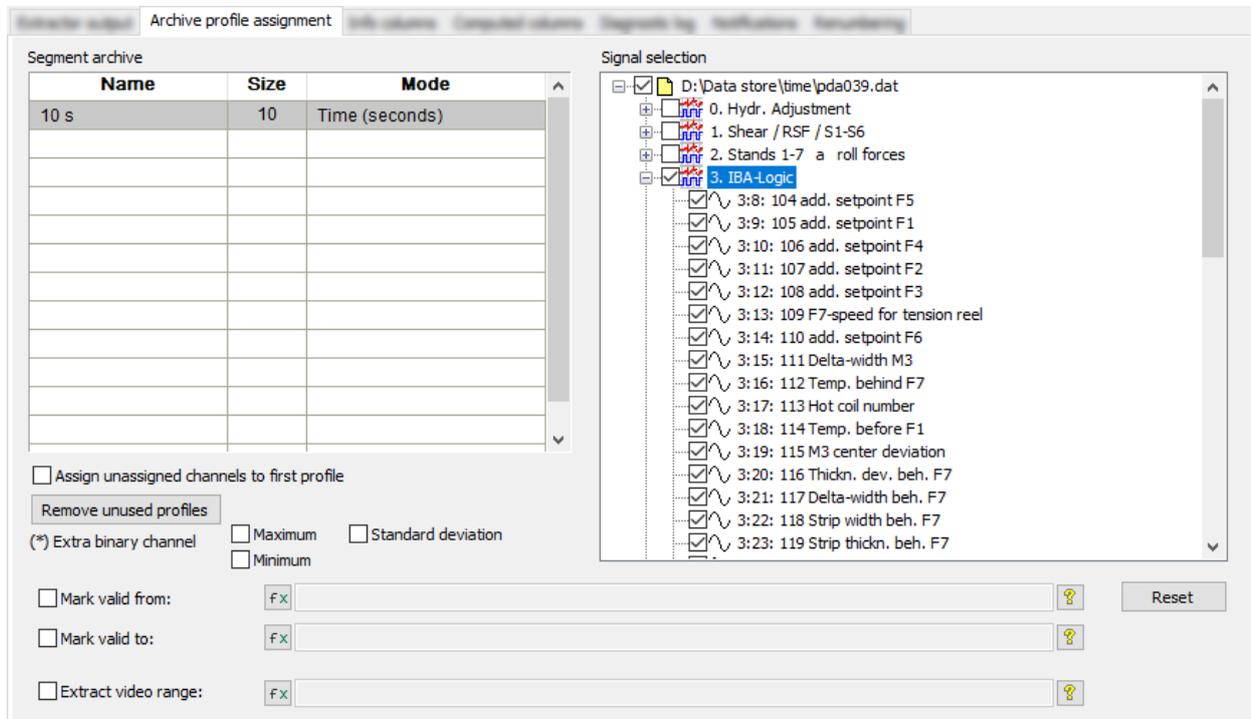


Table columns

Name column

Select a cell in the *Name* column and enter a profile name.

It is recommended to use a name that reflects the selected sampling rate for the extracted file, e.g. "10 s" = Size (10) × Mode (Time (seconds)).

Mode and Size column

In the *Size* column, enter a multiplier to reflect the sampling rate for the extracted data, e.g., 10 s = "Size (10)" × "Mode (Time (seconds))".

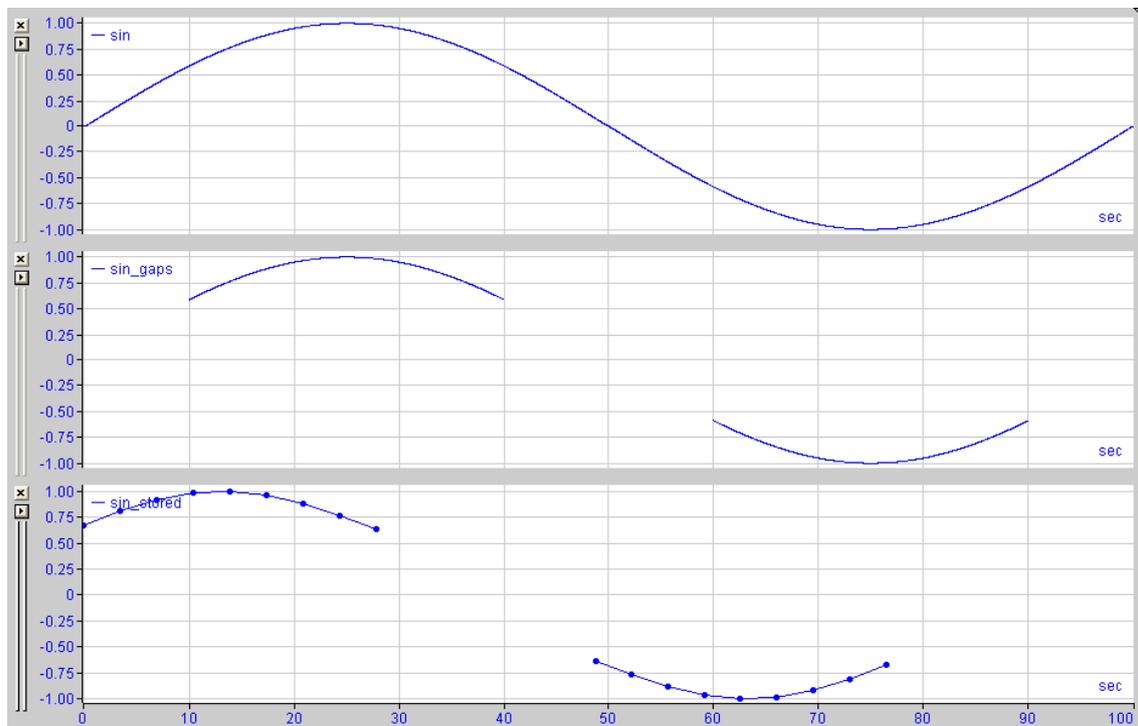
Click in the *Mode* column and select an extraction mode:

- For *Time* and *Length* modes, you can enter a size < 1 to obtain fractions of the unit under *Mode*.
When entering the fraction factor for values < 1, be sure to enter a period instead of a comma as the decimal separator (e.g., "0.5" instead of "0,5").
- The **Signal time base* mode uses the original time base of the signal or a multiple of it. Therefore, fractions are not useful here. For example, a temperature measured every minute does not need to be extracted every 100 ms.

If the time base of the recording signal is 1 ms, the 100 ms profile could also be defined like this: *Size* = 100 and *Mode* = *Signal Time Base.

- With the modes *Time segments* or *Length segments*, you can define a fixed number of equidistant time segments or length segments. For example, *Size = 1* and *Mode = Time segments* means that only 1 segment value is stored for the entire data channel.

If parts of the source signal are invalid, the extracted segments containing the invalid parts are empty. If the empty segments are at the beginning or end of the signal, they are not included in the extracted data. If the empty segments are located between valid segments, the extracted data contain corresponding gaps with invalid values.



- *Time, non-equidistant* mode is supported only for extraction in text files and extracts non-equidistant data.

Assign unassigned channels to first profile

If you enable this option, all signals that have not been manually assigned to a profile are automatically extracted with the first profile. However, they are not marked in the signal assignment. This happens e.g. by the extension of the data files in *ibaPDA*. This option can be useful if the file structure changes over different files.

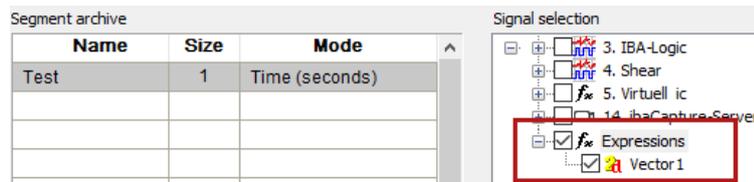
<Remove unused profiles>

Use this option to automatically delete profiles which have no signals assigned.

Signal selection

You have to assign signals to the created profiles. Select signals to be included in the extracted data. The extracted signal values are the average values of the source signal values within the selected sampling rates.

When you enable or disable a module node, you also enable or disable all signals that belong to that module. You can enable or disable single signals in the opened module node. The selected signals are always assigned to the marked profile (gray background color). When you change the profile, the associated selected signals also change. You can assign only 1 profile to a signal.



Note



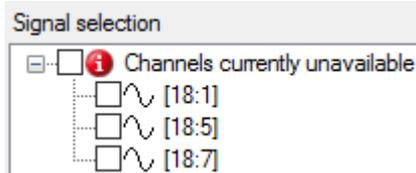
If you assign a length mode to a time-based signal or a time mode to a length-based signal, then an error message appears during extraction.

The **Signal time base* mode is suitable for time-based and length-based signals.

Note



If the archive profile specifies signals that are not present in the current analysis file, they are listed under *Channels Not Currently Available*.



Extra binary channel Maximum, Minimum or Standard deviation

If you enable this option, subchannels are created, each representing the maximum, minimum or standard deviation of the source signal values within the selected sampling rates. This option can be useful for newly acquired signals.



Note

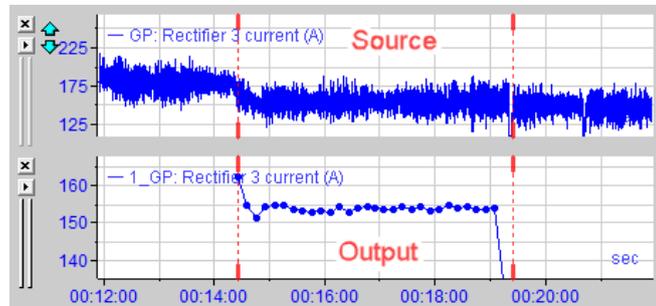


If the selected sampling rate of the profile is identical to the sampling rate of the source signals, these values have no meaning.

Mark valid from/Mark valid to

If you do not want to extract the signal over the full duration, you can define the start and end of the profile range here. You can use expressions to calculate the range from your data.

<input checked="" type="checkbox"/> Mark valid from:	f x	XMarker1()
<input checked="" type="checkbox"/> Mark valid to:	f x	XMarker2()



If neither of the two expressions is defined despite activated options, the signal is extracted completely.

If the specified expressions cannot be evaluated, an error message appears during the extraction.

Extract video range

When exporting videos, only parts of the video may contain important information. By providing a corresponding digital signal, only the parts of the video marked by this signal are extracted to the file. This procedure helps to save storage space.

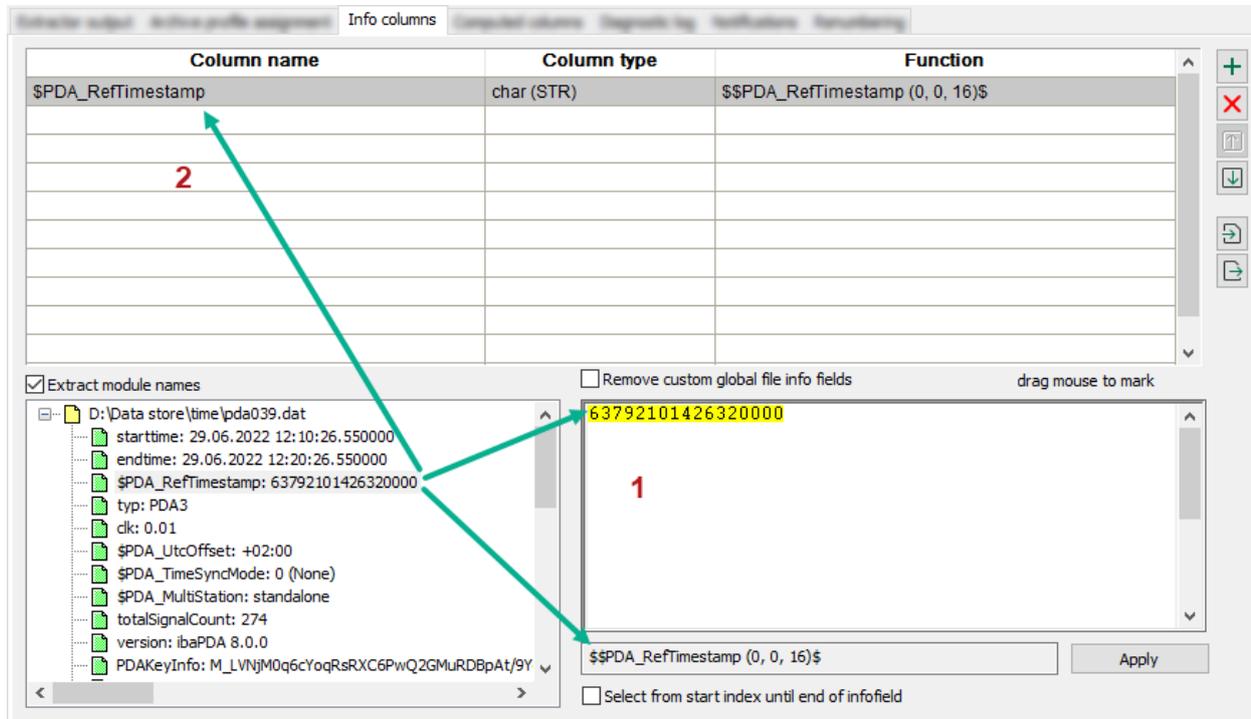
<Reset>

Remove all signals from the selected profile.

3.3 Info columns

Use the *Info columns* tab to extract information from the info part (e.g. start time, texts) of the source file. This data can then be found in the info part of the extracted files.

You have two possibilities to create info fields.



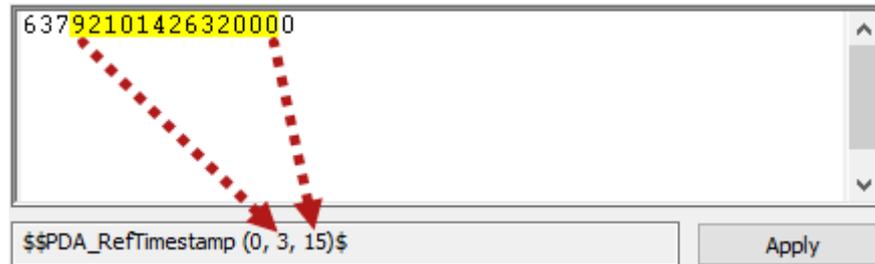
- Click on the source field. The field information is transferred to the processing field (1). Select the range of characters with the mouse and click on <Apply>. Your selection is transferred to the table (2) and the processing field. The *Function* column shows the character selection.
- Double-click on the source field. The field information is completely transferred to the table (2) and the processing field (1).

You can rename the *Column name* manually. *Column type* specifies the data type and is always "char" for file extraction. You can change the character range of the *Function* in the processing field.

Select group

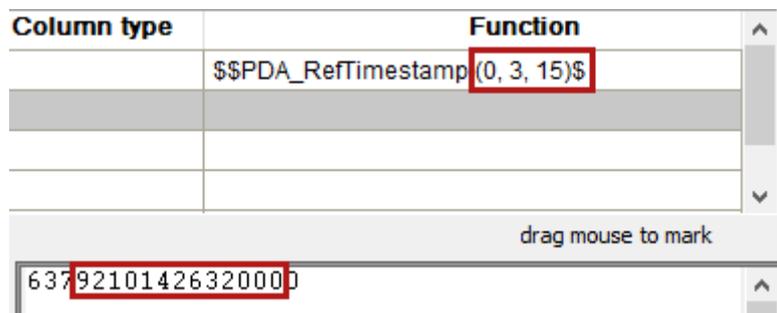
You can also select a group of characters from the contents of the info field.

Click the first required character, drag the selection to the last required character.



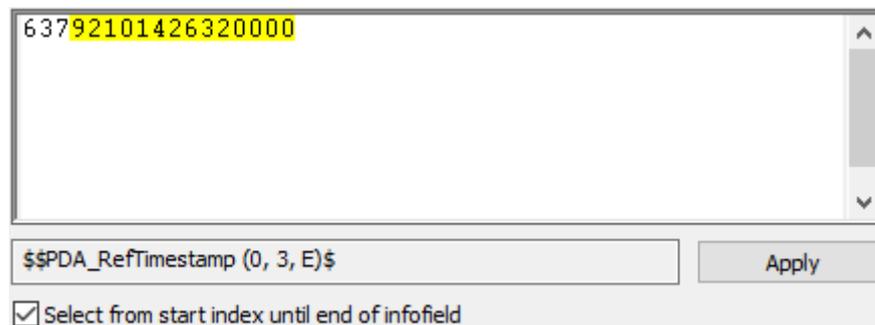
Click on <Apply>.

→ The excerpt from the source string is transferred to the column field.



Select from start index to end of info field

If you enable this option, all characters from the selected start point to the end of the info field are selected.



Remove custom info fields

By default, *ibaAnalyzer* also extracts custom info fields (formerly called technostring). However, these custom info fields can contain sensitive information. If you enable the *Remove custom info fields* option, these info fields are not extracted, but only the info fields that you have defined under *Info columns* or *Computed columns*.

Extract module names

This option is relevant only for the extraction to databases.

3.3.1 Modify info columns

Using the buttons on the side you can change the order of info columns or import or export information about the columns.

	Add line	Adds a new empty line above the currently selected line.
	Delete line	Deletes the currently selected line.
	Move line up/down	Moves currently selected line up/down.
	Import info file	Import info column information from text file. The "Open text file" dialog appears. Alternative: Right-click on the table header and select <i>Import</i> .
	Export info file	Export info column information to text file. The "Save text file" dialog appears. Alternative: Right-click on the table header and select <i>Export</i> .

<Create Logicals>

At different places in *ibaAnalyzer*, you can define and manage logical expressions. This button creates a group of logical expressions from the defined computed columns. You can then manage the created logical expressions via the *Logical signals* dialog. When renaming or moving the expressions, all references to them are automatically adapted.

If you have already used the same expressions or the same names elsewhere in *ibaAnalyzer* and they are not yet grouped, you can select how you want to proceed with these expressions: ignore, overwrite other expression or replace this expression with the other one.

Special feature text signals

You can also use text signals in computed columns. A column with the name of the text signal and its first value is then created in the file.

If you want to extract a specific value from a text signal with several values in the data file, you have to select this value explicitly using a suitable signal expression.

Example: Selection of the third value (counting starts at 0)

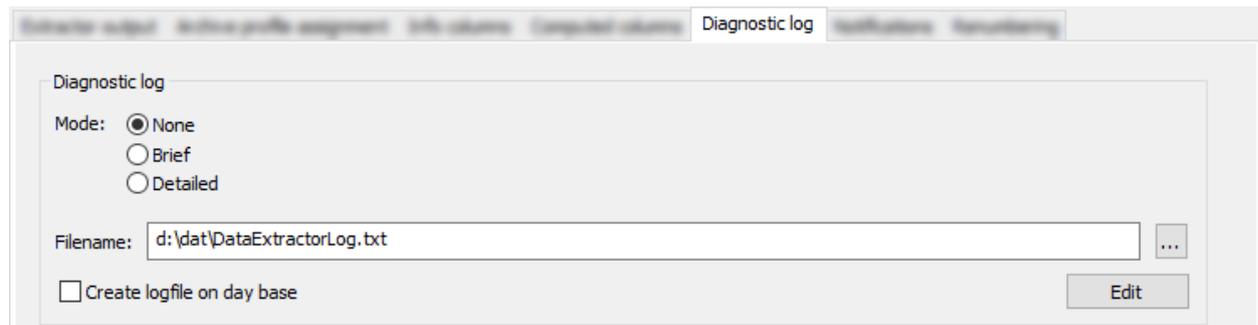
```
"XCutRange([text channel], XFirst([text channel],2), XFirst([text channel],2) + 0.01)"
```

3.5 Diagnosis and notification

In a production environment it is necessary to monitor automated processes, to log processing information for diagnostic purpose and to generate notifications.

3.5.1 Diagnostic log

The diagnostic log is useful for checking progress when automatic extraction is operating (e.g. triggered by the *ibaDatCoordinator*). If you select a log mode, *ibaAnalyzer* creates a log file and updates it with each extraction.



Mode

- **None:** *ibaAnalyzer* does not create log files of the extraction.

- **Brief:** The log file only contains the filenames of the extracted files.

```
24-May-17 13:45:54: Start extract file 'D:\IBA\dat files\dat-training\pda_training021.dat'
24-May-17 13:46:57: Start extract file 'D:\IBA\dat files\dat-training\pda_training021.dat'
```

- **Detailed:** The log file contains the filenames of the extracted files and individual channels.

```
24-May-17 13:16:10: Start extract file 'D:\IBA\dat files\dat-training\pda_training021.dat'
24-May-17 13:16:10: Start extraction to file 'd:\dat\pda_training021_de.dat'
24-May-17 13:16:10: Start extract channel 18:1 GP: Rectifier 1 current with profile 100ms
24-May-17 13:16:10: Start extract channel 18:3 GP: Rectifier 2 current with profile 100ms
24-May-17 13:16:10: Start extract channel 18:5 GP: Rectifier 3 current with profile 100ms
24-May-17 13:16:10: Start extract channel 18:7 GP: Rectifier 4 current with profile 100ms
24-May-17 13:16:10: Extract completed
24-May-17 13:16:35: Start extract file 'D:\IBA\dat files\dat-training\pda_training021.dat'
24-May-17 13:16:35: Start extraction to file 'd:\dat\pda_training021_de_00.dat'
24-May-17 13:16:35: Start extract channel 18:1 GP: Rectifier 1 current with profile 100ms
24-May-17 13:16:35: Start extract channel 18:3 GP: Rectifier 2 current with profile 100ms
24-May-17 13:16:35: Start extract channel 18:5 GP: Rectifier 3 current with profile 100ms
24-May-17 13:16:35: Start extract channel 18:7 GP: Rectifier 4 current with profile 100ms
24-May-17 13:16:35: Extract completed
```

Filename

Enter the path and the name of the log file. If you did not select a daily log, the default filename is e.g. [DataExtractorLog.txt](#).

Use the <...> button to browse the file system if necessary.

Create logfile on day base

ibaAnalyzer creates a log file for each day, e.g. [DataExtractorLog_23_02_2023.txt](#).

<Edit>

Open the selected log file in Notepad and edit it as needed.

Note

If logging is activated permanently, you have to implement a cleanup strategy for the log files externally (not part of *ibaAnalyzer*).

3.5.2 Notifications

On the *Notifications* tab, you can choose between two communication methods how to receive notification or record events.

E-mail address

Enter the address and select the send mode.

Write to Windows application event log

Events are shown in the Windows Event Viewer. Select the send mode.

Level	Date and Time	Source	Event ID	Task Category
Information	24-May-17 14:40:43	ibaAnalyzer	1	None
Information	24-May-17 14:39:41	ibaAnalyzer	1	None
Information	24-May-17 14:38:52	ibaAnalyzer	1	None
Information	24-May-17 14:36:56	ibaAnalyzer	1	None
Information	24-May-17 14:17:11	ibaAnalyzer	1	None

Event 1, ibaAnalyzer

General Details

Data extract file D:\IBA\dat files\dat-training\pda_training021.dat ok

Send modes for each communication method

- **On success**
Notification when extraction is successful.
- **On failure**
Notification when any extraction has failed.
- **On failure (1st failure only)**
Notification when first extraction has failed.
- **On completion**
Notification made when the extraction has been completed, success or failure is irrelevant.

3.6 Renumbering

If multiple files are open and you want to extract multiple signals from multiple files, it is likely that some channels have the same ID number. The channels must therefore be renumbered, otherwise they cannot be uniquely identified in the exported media. *ibaAnalyzer* can perform this renumbering automatically, but will issue a warning if this is required before proceeding with the extraction.

The channel IDs consist of a module number and a position within the module.

The *Renumbering* tab allows you to specify an offset to the module numbers for each file, so you can prevent ID collisions and avoid automatic renumbering.

Global module offset for channel:

Module offset per file:

	Active	File	Expression
1	<input checked="" type="checkbox"/>	D:\dat\pda_training021_de.dat	<input type="text" value="0"/>
2	<input checked="" type="checkbox"/>	D:\dat\pda_training021_de_02.dat	<input type="text" value="1000"/>
3	<input checked="" type="checkbox"/>	D:\dat\pda_training021_de_01.dat	<input type="text" value="2000"/>
4	<input checked="" type="checkbox"/>	D:\dat\pda_training021_de_00.dat	<input type="text" value="3000"/>
5	<input type="checkbox"/>		<input type="text" value=""/>

Suffix for expressions:

Global module offset for channel

This number is added to the data file channel numbers.

File column

The open data files are listed automatically in the File column.

Expression column

Enter the offset for each data file in this field.

Active column

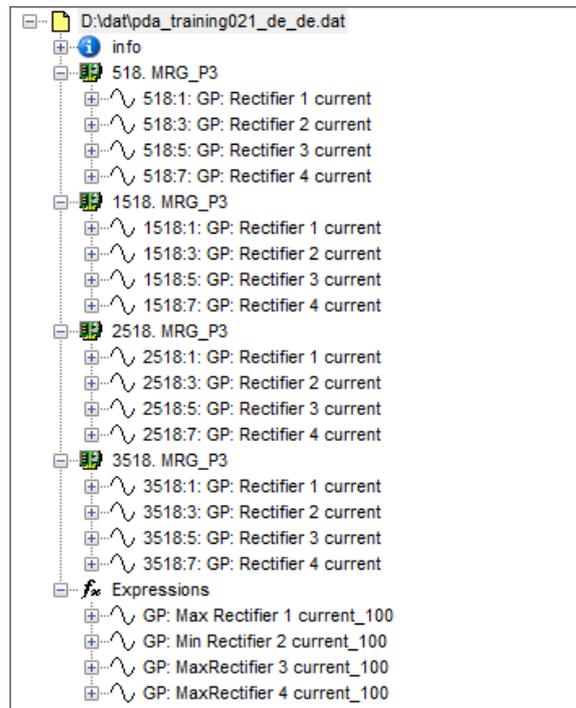
Only active files receive a channel offset. Extraction of channels in inactive files results in a notice.

Suffix for expressions

This applies only to expressions, names and comments created in *ibaAnalyzer*. The specified suffix is appended to the expression names.

Examples for extracted file

Extracted data with renumbering (based on the settings above) are displayed in the extracted file as follows.

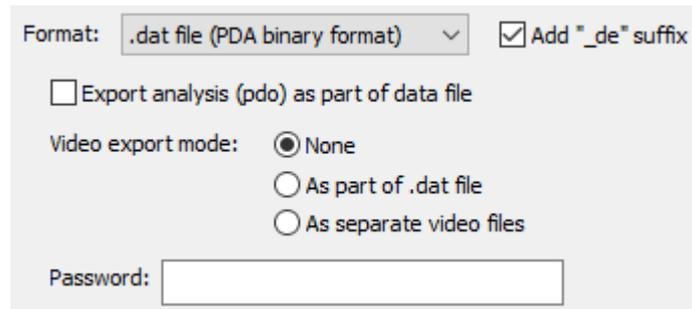


4 Supported file formats

With the file extraction function you can convert available data into various formats. The functionality is constantly extended to support other formats. Currently, iba data files (*.dat), ASCII (or CSV), COMTRADE, TDMS, Apache Parquet and Matlab (*.mat) are supported.

4.1 iba format (.dat file)

If you select the format *.dat file (PDA binary format)*, you can create new and modified data files.



The screenshot shows a dialog box for exporting data. At the top, the 'Format' dropdown is set to '.dat file (PDA binary format)' and the 'Add "_de" suffix' checkbox is checked. Below this, there is an unchecked checkbox for 'Export analysis (pdo) as part of data file'. Under 'Video export mode', the 'None' radio button is selected, with 'As part of .dat file' and 'As separate video files' as unselected options. At the bottom, there is a 'Password:' label followed by an empty text input field.

If the option *Export analysis (pdo) as part of data file* is selected, the exported data file contains the analysis. When such a file is opened again by *ibaAnalyzer*, the decision can be made whether to import the embedded analysis.

Note



All extracted data is converted to Real format. This can cause a loss of precision for some data types.

You can also create password-protected files. The password specified in this dialog is used for all exported files.

Video export mode

If necessary, you can additionally embed available video data in the exported data files. You have the following options for this:

- *None*
No video data are exported.
- *As part of .dat file*
ibaCapture videos can be exported as part of a new data file. All videos that are selected in the signal tree are exported.
- *As separate video files*
ibaCapture videos are exported as separate MP4 files to the same directory as the data files. Although 2 separate files are created, opening the resulting data file automatically calls up the associated video file. The filename suffix "_nn" indicates the individual pairs.

Note

For exporting the videos, *ibaAnalyzer* needs access to the video files or the *ibaCapture* server.

4.2 Text file (CSV)

If you select the option *.txt file (CSV)*, you can export to plain text files.

ibaAnalyzer automatically creates separator-based text files. This means that all signal data are organized in columns and the columns are separated by the specified separator in the extracted file. You can customize the resulting files using various settings.

Code page

You can select different encodings for the resulting file. It is recommended to use UTF-8 as a standard.

Decimal character

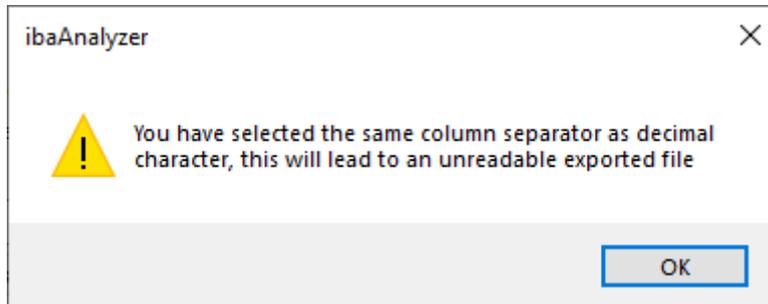
Select if a comma or point is used as decimal separator for the exported values. The *System* option applies the standard used by the current operating system.

Column separator

The default option is to use a tab as separator of the individual columns. You can select other separators, e.g. to create CSV files (with comma separated values).

Note

If you set *Comma* as decimal separator and as column separator, the output files are not usable. *ibaAnalyzer* displays a corresponding warning message if you have set this combination.

**Time export mode**

Select if time data need to be exported. If so, select the format. The time stamp appears as a separate column in the exported file.

- *None*
No timestamp data are exported.
- *Relative time offset*
Begins with "0" and counts the seconds from the file start.
- *Absolute date/time*
Displays the absolute date and time.

Time	[18:1]	[18:3]	[18:5]
30.01.2017 00:11:55.120000	0	217.272	190.276
30.01.2017 00:11:55.220000	0	238.018	196.484
30.01.2017 00:11:55.320000	0	219.696	182.417
30.01.2017 00:11:55.420000	0	226.913	182.002

- *Absolute ISO date/time*
Date and time are extracted in ISO 8601 format.
`YYYY-MM-DDThh:mm:ss0000+hh:mm`

Time	[18:1]	[18:3]	[18:5]
2017-01-30T00:11:55.120000+01:00	0	225,451	186,9
2017-01-30T00:11:56.120000+01:00	0	232,631	184,234
2017-01-30T00:11:57.120000+01:00	0	238,449	187,122
2017-01-30T00:11:58.120000+01:00	0	235,515	182,911

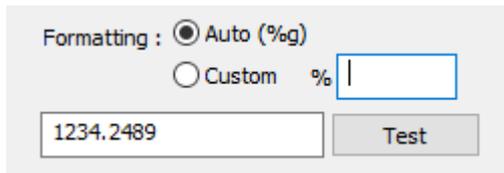
Formatting

You can specify the number of digits used when writing floating point values to a file. The syntax follows a printf statement in C++.

The *Auto* option uses the format *%g*. Under *Custom* you can use different variants. Use a suitable reference for printf in C++.

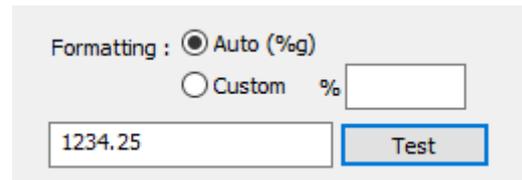
In order to test the behavior, an additional test field is available which formats the input to the corresponding format.

Enter value and click on <Test> (left image). Then, the formatted value is displayed (right image).



Formatting : Auto (%g)
 Custom % |

1234.2489 Test



Formatting : Auto (%g)
 Custom % |

1234.25 Test

Export signal name, Export signal unit

If you enable this option, an additional line is added at the header of the resulting file. The lines contain either the name or the unit of the measured values.

Export infofields and computed columns

This option permits the extraction of all signals created in the *Info columns* tab and the *Computed columns* tab of the Data extractor window. The data is written into a separate part of the file with a key-value structure.

Note



Text signals in the source file are treated like any other source signal.

Skip leading/trailing empty rows

Normally, the exported data starts with the first line at the start time and ends with the last data corresponding to the end time of the file.

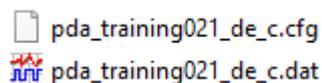
If not all data was extracted (e.g. because XMarkValid was applied), this may result in complete lines with "NaN" in the result.

If you enable these options, such empty lines are automatically removed from the resulting text file.

4.3 COMTRADE

The *IEEE Standard Common Format for Transient Data Exchange for Power Systems* (short COMTRADE) defines a particular format for the exchange of data files as documented in the IEEE Std C37.111-1999 standard. The standardization applies to both the format of the files and the type of media to be used for exchanging fault signal, test data or simulation data of energy supply systems.

During the COMTRADE export, *ibaAnalyzer* generates a DAT file with the measured values and a CFG file with the configuration data such as channel information (signal number, signal name, info columns), start time, end time, etc.



Type

ibaAnalyzer supports different types of this format, which you can select via the selection list.

According to the COMTRADE convention, the information regarding Station name and Recording device must be added to the file. This is done by entering the information in the corresponding fields. This information is stored in a CFG file that *ibaAnalyzer* creates during the export process in addition to the DAT file.

Net Frequency

Set the appropriate main frequency (50 Hz or 60 Hz) here.

4.4 TDMS

Using this option, you can extract to the binary TDMS file format by National Instruments (Technical Data Management Streaming). You can open these files in e.g. LabVIEW and DIAdem.

Note



You have to install additionally *ibaTDMSBundle* to unlock this feature. The installation file [ibaTDMSBundleInstall.exe](#) is available on the data storage medium "iba Software & Manuals" and in the download area on the iba website. For further information, contact iba support.

4.5 Apache Parquet

Apache Parquet is a column-oriented, binary data format which provides efficient data compression and different encodings. Due to its columnar structure and the possibility to add meta data to the file, it resembles the iba data file format. Also due to the comparable storage size, it is recommended to use these files as an interchange format to external systems.

The data in the output file are structured as follows:

- A channel (or expression) corresponds to a Parquet column
- The module structure available in iba data files has no direct pendant in Parquet and is therefore mapped using meta data (see below).
- All info fields are stored as Parquet meta data.

Note



All extracted (numerical) data is converted to the FLOAT Parquet data type. This can cause a loss of precision for some data types. STRING and BOOL receive the same corresponding Parquet data type.

Extractor output settings

Compression

The Apache Parquet format offers different compression methods. *ibaAnalyzer* supports plain encoding (Uncompressed), Snappy, Gzip, Brotli, LZ4 and ZStandard.

Row group size

The Apache Parquet format has an additional row-wise structure mechanism called "column chunks" or "row group". In *ibaAnalyzer* you have the possibility to choose the row group size according to your needs. The input corresponds to the number of rows per row group.

Create Spark compatible files

You can use the Apache Spark framework to work with Parquet files. Since several characters are not allowed as column name within this framework this option replaces all such characters by underscores.

Time export mode

Select if time data needs to be exported. If so, select the format. The time stamp appears as a separate column in the exported file.

- *None*
No time stamp data are exported, however the start time and sampling rate are still available as meta data.
- *Relative time offset*
Begins with "0" and counts the seconds from the file start. The additional column contains FLOAT values.
- *Absolute Date/Time*
The additional column contains the absolute date and time. The Parquet data type TIME-STAMP is used.

Derive column names from

Here, you can select the column names for the Parquet file. The channel numbers, channel names or one of the comments are available for selection. Note that when using a comment, this information needs to be available.

When selecting the channel number, the names are formatted as follows:

- [M:C] for analog channels
- [M_C] for digital channels (dots are not allowed for channel names in the Parquet format)
- [M:C:S] for subchannels

With "M" as the module number, the signal (or channel) number is "C" and the subchannel number is "S".

The Parquet format does not allow non-unique column names. If this is the case in the original data, *ibaAnalyzer* automatically adds a corresponding suffix like "_1", "_2", etc.

For the extraction of *ibaQDR* data files, you can optionally add the measurement location number as prefix before the column name of the channel number.

Meta data

If data are exported from iba data files, several meta data (or info fields) are available. These data are also written to the Parquet format, where only one level of meta data exists. Therefore, these data are structured as follows:

- File level information (the standard info fields) are stored as normal key value pairs with the info field name as key.
- For computed columns and info columns the specified name is used as key
- Module level information use a key of the form "M[x]y" where "x" is the module number and "y" the field name.
- Channel level information use a key of the form "[x]y" with "x" being the channel name and "y" the field name.

This structure of data also enables *ibaAnalyzer* to restore the complete file structure, when opening the extracted Parquet file.

Length-based data and ibaQDR data

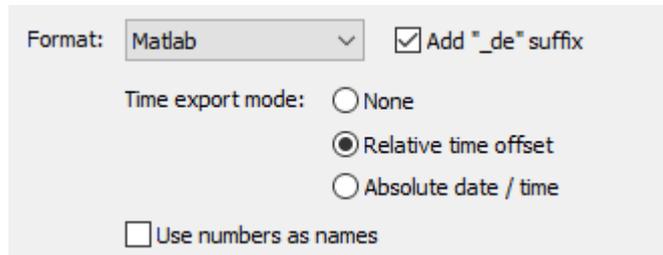
You can also extract length-based data or *ibaQDR* files to Parquet format.

If the correct length-based profile is used for the extract, *ibaAnalyzer* writes additional meta data to the file to indicate this. The following fields are used:

- "Lengthbased"
This field indicates that the column contains length-based data.
- "LengthBase"
In case of length-based data this field shows the sample rate in meter.

4.6 Matlab

The software Matlab distributed by MathWorks provides its own (binary) data format with file extension ".mat". In order to better support the Matlab integration, *ibaAnalyzer* can create MAT files. You can open these files directly in the Matlab software.



Time export mode

Similar as for other formats, it is possible to export an array containing the timestamps either as relative time offset or as formatted string. The time array is added to the "fileinfo" structure which is described below.

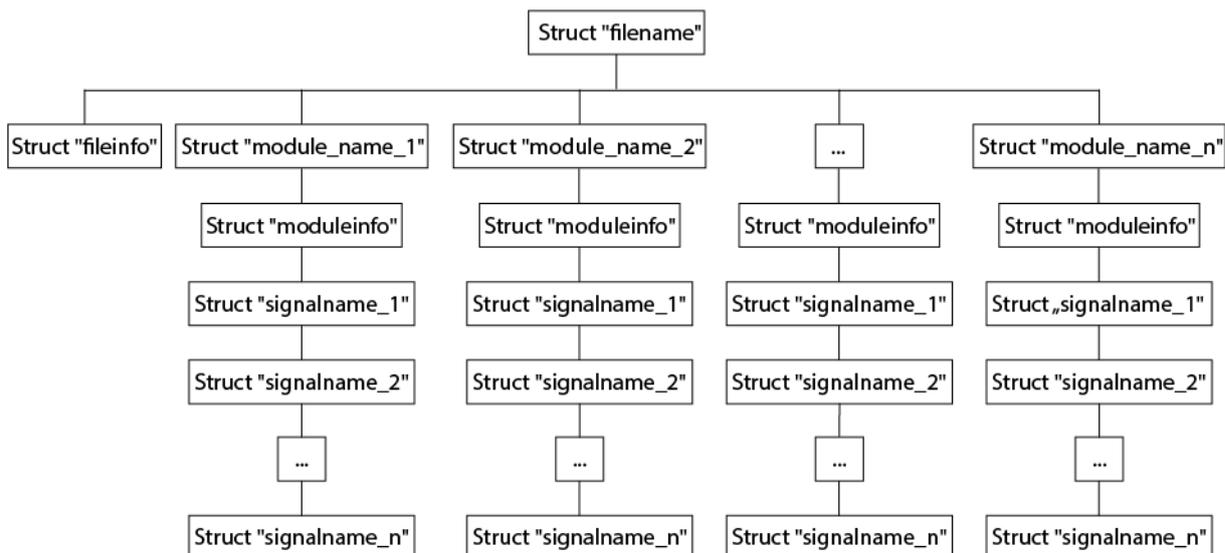
Use numbers as names

With this option it is possible to not use the module and signal names for the names of the struct objects (see below). In this case the module or signal number is used with some prefix "M" for modules and "S" for signals.

In order to display the correct name when opening the file again with *ibaAnalyzer*, the original names are stored in an additional field "name" within the structure.

Data structure

The software Matlab supports different data types and structures. *ibaAnalyzer* supports MAT files which contain so-called struct objects. *ibaAnalyzer* creates nested struct objects in the following form:



This structure is used to resemble the internal structure of the iba data files.

The struct "filename" is the root structure. It contains only other structures containing info fields or modules. The name of the structure is derived from the filename displayed in the *ibaAnalyzer* signal tree.

The file-level info fields are stored as key-value pairs in a fixed structure "fileinfo". The field "clk" contains the sampling rate of the signal data. Further, a field "starttime" contains time stamp of the first data point. Several other info fields will automatically be added to this structure.

The individual modules are present in form of structures having the name of the module. In case the *Use numbers as names* option is set, the structures are named as "Mx" where x is the module number.

Every module has a fixed structure "moduleinfo" which contains module-level info fields as well as a field "name" containing the module name, and a field "ModuleID" containing the module number.

Like for the modules, the individual signals within one module are present as structures using the signal name or a string "Sx" with "x" being the signal number.

Every signal structure contains an array "data" which holds the signal values. Other key-value pairs are written representing the signal-level info fields. The most important fields are:

- "SignalID"
the signal number within the module
- "Name"
the signal name (will be displayed in *ibaAnalyzer*)
- "Unit"
the unit (if present) as displayed in the signal grid
- "PDA_Comment1"
the comment 1 displayed in the signal grid. Similar for comment 2
- "PDA_TBase"
This field is present in case the sampling rate is different from the global "clk" value.
- "Lengthbased"
This field indicates that the column contains length-based data.
- "LengthBase"
In case of length-based data this field shows the sample rate in meter.

Naming conventions

When extracting to MAT files, the nested structs are given names automatically. If you enable the *Use numbers as names* option, the signal ID is used.

5 Extraction via the command line

Note



To use the command line switches for extraction, you need the *ibaDatCoordinator-File-Extract* license.

If you run *ibaAnalyzer* via the command line, some of the command line switches for *ibaAnalyzer* are especially suitable for file extraction. This allows you to automate post-processing. The used syntax is the following:

```
ibaAnalyzer.exe datfilename1 [datfilename2] ... [datfilenamen]  
[pdofilename] [/switch]
```

ibaAnalyzer then opens with various data files, an analysis and various options, which are described below. In each case, *ibaAnalyzer* automatically opens all specified data files and executes the assigned tasks. Note that in some cases the analysis is mandatory, for example if the information required for the task is only available in the analysis file.

The following command line switches are useful for file extraction. For a detailed description of the switches, see Part 1, *Starting with Command Line*.

- /extract[:filename]
- /append
- /print
- /report[:filename]

6 Automatic extraction using ibaDatCoordinator

With the *ibaDatCoordinator* tool and the *ibaDatCoordinator-File-Extract* license, you can automate the data extraction into other file formats.

The screenshot shows the configuration window for automatic extraction. Key settings are highlighted with red boxes:

- Target:** Extract to file: (The other option is 'Extract to database').
- File type:** Binary (.dat) (Other options: ASCII (.txt), COMTRADE (.dat;.cfg), TDMS (.tdms;.tdms_index), PARQUET (.parquet), Matlab (.mat)).
- Subdirectories:** Each day (Other options: Each hour, Each week, Each month, Keep original structure, None).
- Cleanup strategy:** None (Other options: Limit subdirectories to 10, Limit disk space usage to 1024 Mb, Minimal free disk space 1024 Mb).
- Output file settings:** Set 'Date Modified' time of output file to match .dat file (This checkbox is highlighted with a red box).
- Monitor ibaAnalyzer:**
 - Memory limit: abort task if ibaAnalyzer starts using more than 2048 Mbytes of memory
 - Time limit: abort task if ibaAnalyzer takes more than 10 minutes to complete

Here, some settings for the extraction function of *ibaAnalyzer* in combination with the Extract task in *ibaDatCoordinator* are described.

Extract analysis file

Specify the path and the filename of the PDO file in this field or select the file using the browser button. To open the analysis in *ibaAnalyzer*, click on the *ibaAnalyzer* button.

Target/Target directory

Select the *Extract to file* option. Specify the path in this field or select the file using the browser button. If username and password are required to access the (remote) directory, you can enter them there as well. You can check the connection using the <?> button.

The *Extract to database* option requires an *ibaDatCoordinator-DB* license.

File type

Select the required file type. The available file formats are described in [↗ Supported file formats](#), page 24

Date modified

For archiving and sorting it may be necessary to manipulate the "date modified" tag of the resulting file. Enable the option *Set 'Date Modified' time of output file to match .dat file* to have the same modification date as in the original file.

Other documentation

More detailed descriptions of other available options or for the general use of the *ibaDatCoordinator* software can be found in the corresponding manual. For information about extracting data into databases, see part 4, *Database Interface*.

7 Support and contact

Support

Phone: +49 911 97282-14
Fax: +49 911 97282-33
Email: support@iba-ag.com

Note



If you need support for software products, please state the number of the license container. For hardware products, please have the serial number of the device ready.

Contact

Headquarters

iba AG
Koenigswarterstrasse 44
90762 Fuerth
Germany

Phone: +49 911 97282-0
Fax: +49 911 97282-33
Email: iba@iba-ag.com

Mailing address

iba AG
Postbox 1828
D-90708 Fuerth, Germany

Delivery address

iba AG
Gebhardtstrasse 10
90762 Fuerth, Germany

Regional and Worldwide

For contact data of your regional iba office or representative please refer to our web site:

www.iba-ag.com