



ibaPDA

Data Storage

Manual Part 5
Issue 8.14

Measurement Systems for Industry and Energy
www.iba-ag.com

Manufacturer

iba AG
Gebhardtstrasse 10-20
90762 Fuerth
Germany

Contacts

Main office	+49 911 97282-0
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 2026, 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 and can be found in the iba help center docs.iba-ag.com.

Version	Date	Revision	Author	Version SW
8.14	06-2026	New: ibaQDR inspection line data store	rm	8.14.0

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	5
1.1	Target group and previous knowledge	5
1.2	Notations	5
1.3	Used symbols.....	6
1.4	Documentation structure	7
2	Introduction	8
3	Types of data storage	9
4	Datastore configuration toolbar	12
5	Data recording profiles.....	13
5.1	General	13
5.2	Profile types.....	14
5.3	Add and manage data recording profiles.....	15
5.4	"Time" profile	16
5.5	"Time, not equidistant" profile	17
5.6	"Time, DB/cloud" profile.....	20
5.7	"Time/length, ibaQDR inspection line" profile	22
5.8	"Length" profile	23
5.9	"Time, ibaHD" profile	24
6	Configuring data storage	27
6.1	Add a data storage.....	28
6.2	Basic settings for data storage	28
6.3	Trigger mode.....	30
6.3.1	Maximum file time	31
6.3.2	Start trigger.....	32
6.3.3	Stop trigger	37
6.3.4	Trigger pool.....	38
6.3.5	Advanced	40
6.4	Signal selection	41
6.4.1	Assigning a signal to a storage profile.....	43
6.4.2	Checking the profile assignment	43

6.4.3	Disabling the profile assignment	43
6.5	Video triggers	44
6.6	Image trigger	45
6.7	Data file information.....	46
6.7.1	Data file name	47
6.7.2	Data file location.....	49
6.7.3	Subdirectory organization.....	51
6.7.4	Files - Info fields.....	52
6.7.5	Data files - Cleanup strategy	53
6.7.6	Post-processing.....	55
6.8	Apply new data storage configuration.....	55
6.9	Configuration file of data storage	56
6.10	Loading a backup configuration file.....	58
7	Diagnostics for data stores	59
8	Stop prevention	62
9	Certificates.....	64
10	Support and contact.....	65

1 About this documentation

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

1.1 Target group and previous knowledge

This documentation is aimed at 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 they are capable of assessing safety and recognizing possible consequences and risks on the basis of their specialist training, knowledge and experience and knowledge of the standard regulations.

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</i> , <i>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 fully describes the functionality of the *ibaPDA* system. It is designed both as a tutorial as well as a reference document. The sections and chapters essentially follow the procedure for configuring the system.

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

Part 1	Introduction and installation	General notes, license policy and add-ons Installation and program start User interface, system architecture, client server User management, printing
Part 2	I/O Manager	Basic information about the I/O Manager, general settings Groups and vector signals, text signals, outputs, configuration files
Part 3	Data interfaces and modules	Interfaces for the measurement data acquisition Standard interfaces, ibaFOB, Ethernet-based interfaces and more. For interfaces for which there are separate manuals, these are referred to.
Part 4	Expression builder	All functions for calculating virtual signals
Part 5	Data storage	Types of data recording, recording profiles, signal selection
Part 6	Data visualization	All display modes for live data, their operation and settings
Part 7	Appendix	Various additions, error listings, etc.

2 Introduction

Besides the display of data, the storage of measured data is one of the main features of *ibaPDA*. If the measured data are stored and available as data files (*.dat), you can view, analyze or post-process the data conveniently with *ibaAnalyzer*.

The data files generated by *ibaPDA* have a special signature that prevents the measured data being read or post-processed using third-party software. To maintain calculation accuracy, data types such as 32-bit integers or 64-bit floating points (double) are stored in the data file unchanged so that calculations can be carried out in *ibaAnalyzer* with full resolution. The datatype 64-bit floating point is also supported for virtual signals as results of calculations in *ibaPDA*.

The data storage management of *ibaPDA* offers a variety of options which enable you to customize the data storage according to your needs in order to get the maximum benefit.

The dialog for data storage configuration can be opened in the *Configuration – Data Store Manager...* main menu or by clicking on the button in the  main tool bar.

3 Types of data storage

Depending on the licenses registered in the dongle or soft license, different types of data storage are available for configuration in the dialog.

Time-based data store

The data storage occurs with this type in the form of data files (*.dat), which can be evaluated with the program *ibaAnalyzer*. The data files generated in this way can be protected from unauthorized opening via password.

Usually, a (time-based) data storage and a blue link "Add data store" are shown in the data storage tree. With the basic version of *ibaPDA*, you can add one additional data store. You can purchase additional licenses if you should need more data storage:

- **ibaPDA-Data-Store**
This license allows you to add 2 additional data stores per license.
- **ibaPDA-Ultra-Data-Store**
This license allows you to configure up to 255 data stores with an average number of up to 20 signals or less per data store. If required you can configure data stores with more than 20 signals. In this case the number of remaining data stores will be reduced accordingly. For example, if you configure a data store with 110 signals the number of remaining data stores will be reduced by 6.

ibaHD data store

The ibaHD data store is a special type of data storage for recording and visualizing historical data. For this purpose, you need the license and software for *ibaHD-Server*.

For the HD data storage, you can choose between

- ibaHD time-based data store,
- ibaHD event-based data store
- ibaHD length-based data store

In case of a time-based HD data store you can additionally define HD time periods if the connected *ibaHD-Server* supports this feature (*ibaHD-Server* v3.1 or higher). Furthermore, HD time-based data stores can be triggered, e.g. in order to record data only when it makes sense (*ibaHD-Server* v3.2 or higher). You can use signals, time periods or system time for trigger.

In case of an event-based HD data store you'll have to define the events here.

In case of a length-based HD data store you'll have to define the measuring locations here.

In a basic version of *ibaHD-Server*, you will be able to set up 2 ibaHD stores. Licenses for additional ibaHD stores are available.

Other documentation



For more information about ibaHD data storage, please refer to the *ibaHD-Server* manual.

ibaQDR data store

If you use an *ibaQDR* system or have a corresponding license registered, the types *ibaQDR data store* and *ibaQDR inspection line data store* will also be displayed in the "Add data store" dialog. The dedicated data store type for inspection lines contributes to the special requirements of this kind of plants.

It is possible to configure more than one *ibaQDR* data store per system.

A special property of an *ibaQDR* data store is the time and length-based storage of measured values.

Other documentation

For more information about *ibaQDR* data store, refer to the manual for *ibaQDR*.

DB/cloud time-based data store

If you have a license for a database or cloud memory, such as *ibaPDA-DataStore-SQL-Server*, then you can use this form of data storage. iba offers different licenses graded by the number of signals.

The measured values are written (streamed) directly into a suitable database or cloud. The data is always written in a time-based manner.

The following database/cloud systems are supported: Oracle, SQL-Server, MySQL, PostgreSQL, SAP-HANA, Azure SQL and MariaDB.

Other documentation

You will find additional information about DB/cloud data store in the manual on the corresponding product, such as *ibaPDA-Data-Store-SQL-Server-xxx*.

Kafka cluster time-based data store

If you have a license *ibaPDA-DataStore-Kafka*, then you can use this form of data storage. iba offers different licenses graded by the number of signals.

The measured values are written (streamed) directly into an Apache Kafka cluster. The data is always written time-based.

Other documentation

You will find additional information about Kafka cluster data store in the manual on the product *ibaPDA-Data-Store-Kafka-xxxx*.

MQTT time-based data store and Sparkplug B data store

If you have a license *ibaPDA-DataStore-MQTT* , then you can use this form of data storage. iba offers different licenses graded by the number of signals.

The measured values are written (streamed) directly to an MQTT broker. The data is always written in a time-based manner.

Other documentation

You will find additional information about MQTT data store in the manual on the product *ibaPDA-Data-Store-MQTT-xxxx*.

InfluxDB time-based data store

If you have a license *ibaPDA-DataStore-InfluxDB* , then you can use this form of data storage. iba offers different licenses graded by the number of signals.

The measured values are written (streamed) directly to an InfluxDB server. The data is always written in a time-based manner.

Other documentation

You will find additional information about InfluxDB data store in the manual on the product *ibaPDA-Data-Store-InfluxDB-xxxx*.

Note

Whenever you have changed the configuration of one or more data stores you should close the configuration dialog with the <OK> button. At this time, the recording of data stores which were subject to change will be stopped and closed. Other, unchanged data stores will continue to record. The recording of the changed data stores will restart automatically with the new configuration.

Important: Changes in the data storage configuration will not affect the data acquisition of the server. The server continues the data acquisition even if the recordings are stopped and restarted.

4 Datastore configuration toolbar

The *Data Store Manager* provides the following controls for data store configurations.

Toolbar

Symbol	Description
	New configuration; then choose whether an empty configuration (without data-store, profile <i>As is</i> only) or a new default configuration should be created. The default configuration consists of a pre-configured timebased datastore (data file). All signals in the I/O configuration will be selected and assigned to the profile <i>As is</i> .
	Opens an existing configuration file (XML-file .ds or .zip).
	Opens an existing configuration file (.ds or .zip) in simulation mode. The loaded configuration is not aligned with the current licenses and cannot be applied. The simulation mode only provides for display of any datastore configuration.
	Opens access to the automatically saved backups of the datastore configuration in the path <code>%ProgramData%iba\ibaPDA\Backup</code> on the <i>ibaPDA</i> -Server computer. In a dialog window you can select and load the desired backup. Furthermore, you can use the dialog to delete a backup.
	Saves the current, full datastore configuration in an XML-file. You can select or enter the path and filename in the following dialog. The file extension .ds is added automatically.
 	These buttons are disabled, unless you can move selected datastores. Move the selected datastore by one position up or down in the tree.
 	Go forward/backward buttons: Navigate to the previous or following step in the configuration.

5 Data recording profiles

The data recording profiles are used to determine how the data should be recorded. There are different options depending on the type of data recording (time-based, length-based, DB/Cloud, etc.). Basically, you should first configure the profiles for your planned data stores. Then link the signals to the desired profile.

5.1 General

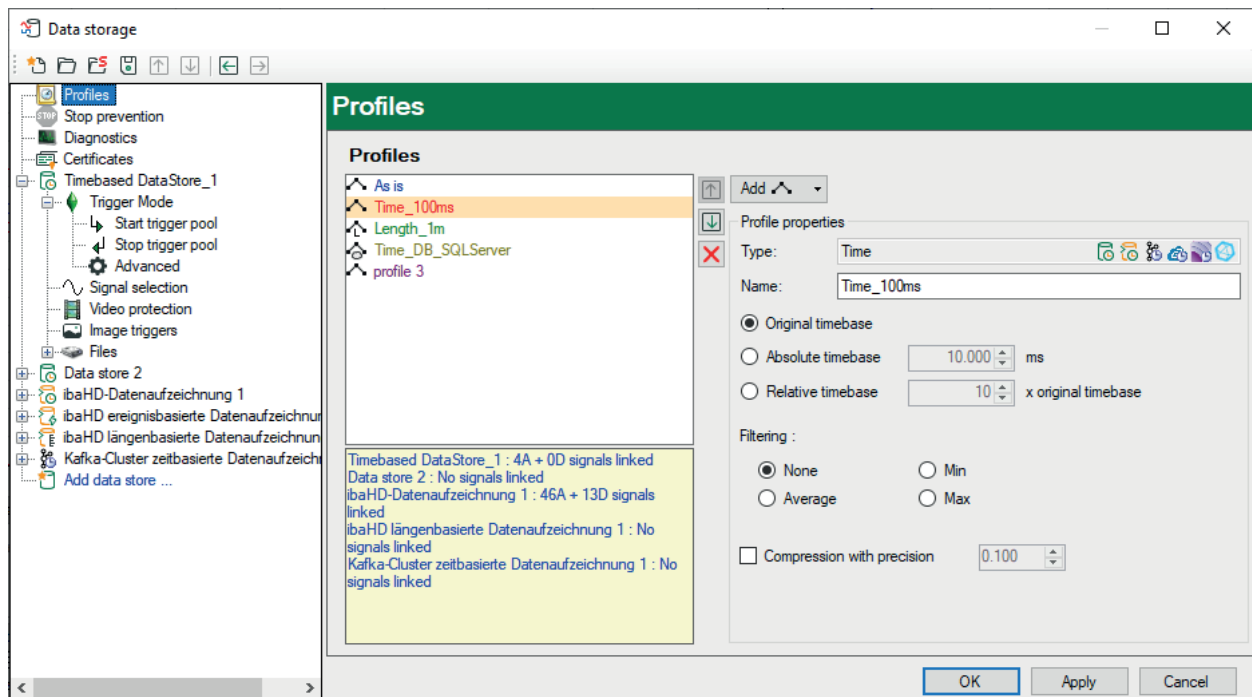
The recording profile function primarily serves to reduce data volumes for time- and length-based data stores. Even when the process phases relevant for the recording have been narrowed down with the trigger settings, still a large number of values is generated that can quickly fill up the storage media. Particularly with general long-term data storage, it is not often necessary when using measured values that they will be available at a later point in time at a high-resolution sampling speed, e.g., 1 millisecond. Additionally, the signal type also determines whether it must be recorded at high or low cycle rates. Rapidly changing signals like currents or forces are usually recorded with a smaller time base than slow-changing signals like temperatures.

Since each signal can be individually linked to a recording profile, it is also possible to differently record signals coming from the same module.

All profile types except the non-equidistant recording have the option of a statistical filter. If the signals are acquired much faster than recorded according to the profile setting, you can use the filter to determine which value (actual value, maximum, minimum or average value) should be saved in relation to the values acquired in the recording interval.

For DB/Cloud data stores, the profile is primarily used to match the target database for ID and value columns.

The following figure shows a typical time-based profile.



Click on the *Profiles* branch in the tree structure at the left of the configuration dialog. In the right-hand part, you can now see a list of available profiles (at least "As is"). Next to the list, you will find more information on the profile you have selected in the list. Below the list, you see for which data storage the selected profile is used and how many signals are assigned to it.

Any number of storage profiles can be defined. Each profile is stored with a unique name assigned by the user, so that the allocation of profiles to the signals is very clear and comprehensible.

5.2 Profile types

ibaPDA offers different recording profiles for the different types of data store. The icon behind the profile name indicates the data store type, which can use this profile.

Profile type	Symbol Profile type	Type of data store	Symbol data store
Time		time-based data store (<i>ibaPDA</i> data file)	
		<i>ibaHD</i> time-based data store	
		Kafka cluster time-based data store	
		MindSphere time-based data store	
		InfluxDB time-based data store	
		MQTT time-based data store	
Time, not equidistant		time-based data store (<i>ibaPDA</i> data file)	

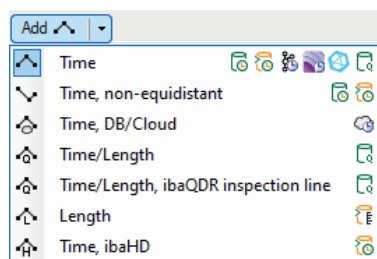
Profile type	Symbol Profile type	Type of data store	Symbol data store
Time, DB/cloud		DB/cloud time-based data store (e.g. SQL-Server)	
Time/length, ibaQDR inspection line		time and length-based data store (<i>ibaQDR</i> data file)	
Length		ibaHD length-based data store	
Time, ibaHD		ibaHD time-based data store with 3 different modes	

Table 1: Profile types and appropriate types of data store

5.3 Add and manage data recording profiles

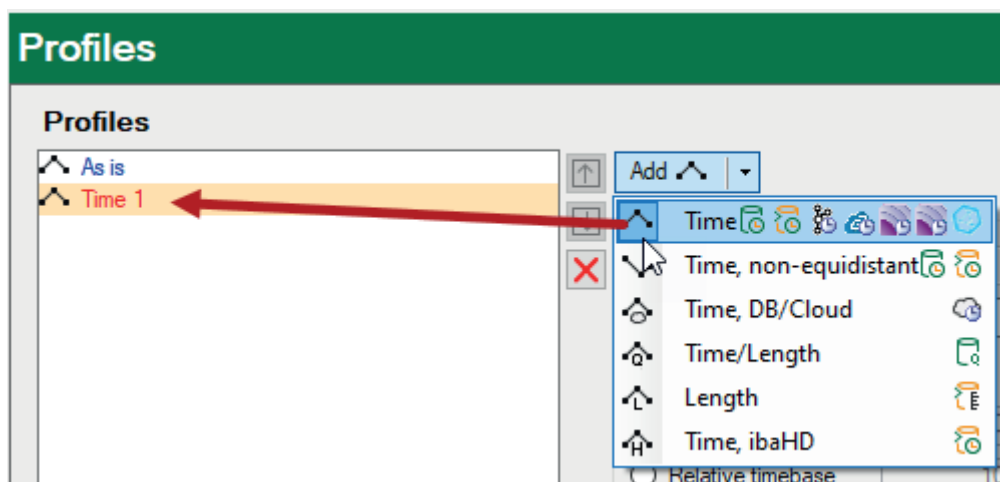
In order to add a new profile, select the *Profiles* branch in the tree structure of the data storage configuration dialog.

In the right window, you will find an <Add> button. An icon on the button indicates the profile type that would be added upon click. If the icon matches the profile type you wish to add, click the <Add> button. If you are not sure about the profile type, just click on the dropdown-list icon on the button in order to show all profile types available.



Select the desired profile type from the dropdown list.

A new profile is added to the profiles list.



The new profile is highlighted with a yellow background.

Then you should define the profile properties (see further below).

Next to the profile list, there are buttons with the following functions:

	Move the selected profile up in the sequence. The icon is not available (gray) if the operation is not possible.
	Move the selected profile down in the sequence. The icon is not available (gray) if the operation is not possible. The "As is" profile is always at the top of the list.
	Delete the selected profile.

5.4 "Time" profile

As the name says, this profile type is used for general time-based recording. The profile stores measured data with equidistant samples. The type is available by default and can be used for the following types of data store:

- Time-based data store
- *ibaHD* time-based data store
- Kafka cluster time-based data store
- MindSphere time-based data store
- MQTT time-based data store
- InfluxDB time-based data store

Profile properties

Type

Display of the profile type with the appropriate types of data store (symbols)

Name

Enter an unambiguous and understandable name here.

Time base

Choose whether you like to record the acquired data with the original time base (as configured under *General settings* in the I/O Manager) or with another time base (absolute time base). If you select *Absolute time base*, please define the desired time base in the field, given in milliseconds.

Upper limit value is 214748 (approx. 3.58 min.)

If you decide to use a *Relative timebase*, enter or set an integer factor to get a multiple of the original timebase. Upper limit is 100,000.

Filter

Select the signal value to be stored with reference to the time base set above.

- None: The actual value will be stored
- Average: The arithmetic average of all measured values, which were acquired in the interval (according to the time base, such as 10 ms or 100 ms) is formed and saved.

- **Min:** Only the smallest value of all samples measured within the defined interval will be stored.
- **Max:** Only the largest measured value of all measured values acquired within the defined interval will be stored.

Compression with precision

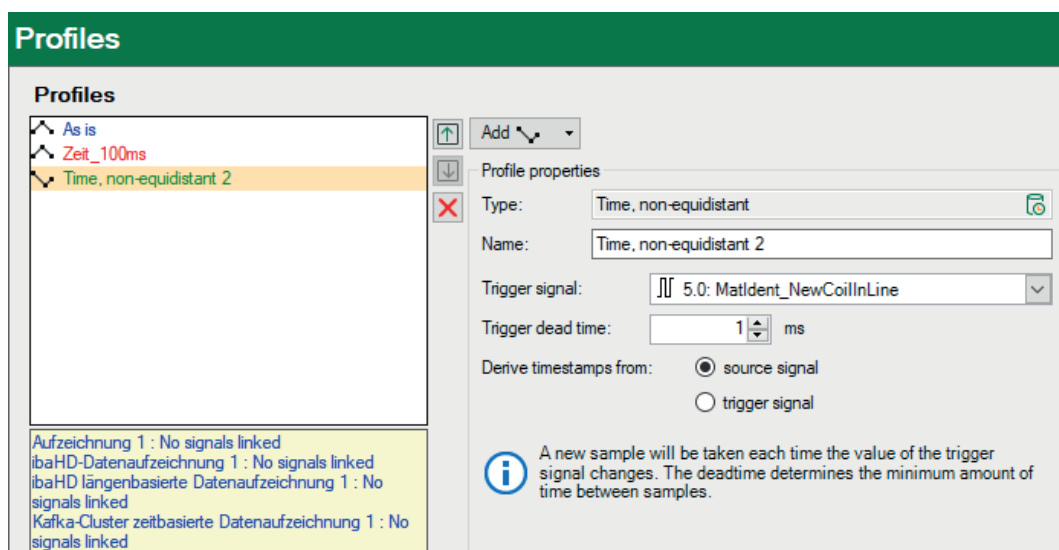
Use this option to achieve a further reduction in data. Method:

every value is subtracted from the previous one. The (absolute) difference is compared to the accuracy value defined in the field nearby. The new sample will only be stored if the difference exceeds the accuracy value (based on the physical unit of the signal). If the difference is equal or less than the accuracy value, an internal counter in the data file will be incremented and the sample will not be stored. Storage space can be saved thereby.

Usually, this method is suitable for signals with a general noise or for sensors with less accuracy. The accuracy value indicates the absolute range given in physical units in which a signal sample can vary without being stored as a new value.

5.5 "Time, not equidistant" profile

This profile type has been developed for specific requirements. It is suitable for situations with an controlled irregular occurrence of measurement data, depending on the process event. It is also suitable, for instance, if you just want "snapshots" of the process data to be triggered by a control signal.



Profile properties

Type

Display of the profile type with the appropriate types of data store (symbols)

Name

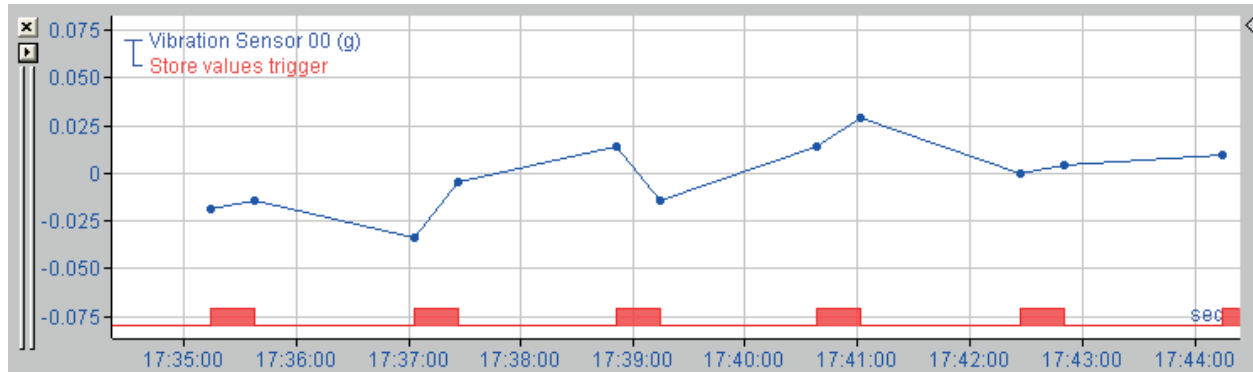
Enter an unambiguous and understandable name here.

Trigger signal

Select a signal from the dropdown list to be used as trigger for writing samples in the data file.

Every time the value of the trigger signal changes, a new sample will be stored.

If you use an analog value for triggering, samples are with high probability recorded continuously unless the analog value remains 100 % constant over periods where no sample should be stored. Using a digital signal as trigger will store samples with every rising/falling edge, as shown in the figure below.



Trigger dead time

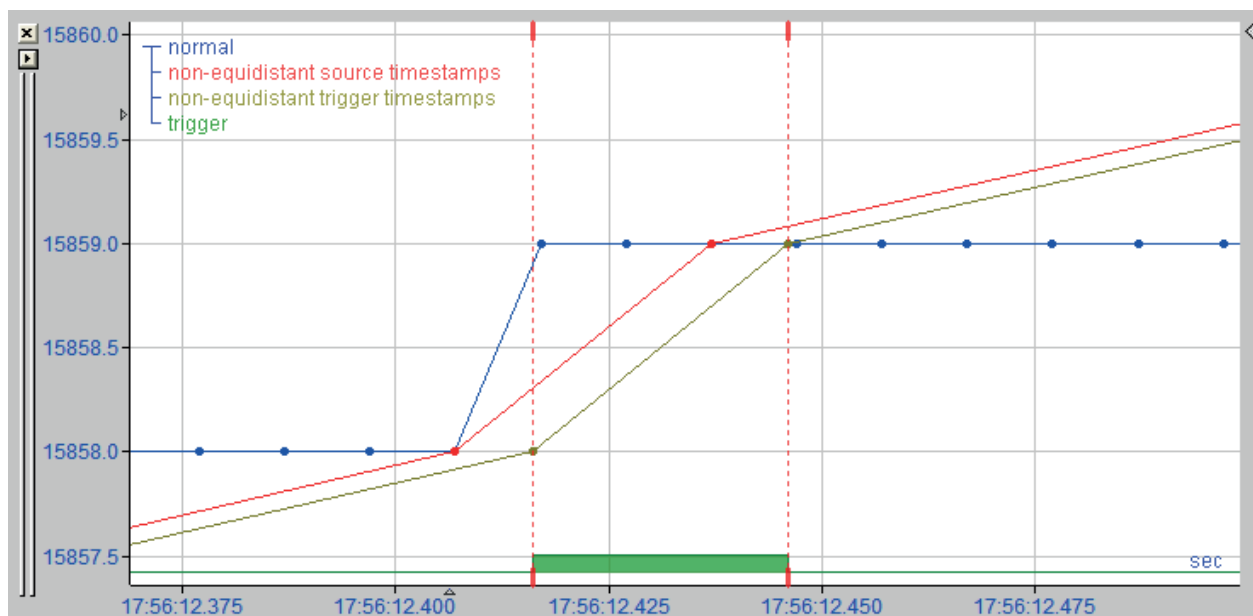
The trigger dead time determines the minimum period between the samples. Trigger events that occur during the trigger dead time are ignored.

Derive timestamps from source signal/trigger signal

If the trigger signal and source signal have different time bases, you can choose which time-stamps to use for the non-equidistant signal. "Source signal" is every signal assigned to this profile.

Explanation

The screenshot below shows another example of the difference between the two options.



The blue signal is the source signal. The green digital signal is the trigger signal. The source signal is sampled at 10 ms. The trigger signal is sampled at 1 ms. You can see that the trigger value changes at a time that does not correspond with the sample of the source signal.

The red signal is the non-equidistant signal that uses the timestamps from the source signal. You can see that the sample points of the red signal correspond with the sample points of the original signal. If you choose source signal timestamps, the timestamp will be smaller or equal to the trigger timestamp.

The brown signal is the non-equidistant signal that uses the timestamps of the trigger signal. You can see that the sample points of the brown signal correspond with the sample points of the original signal moved to the trigger timestamps.

Both options have advantages as well as disadvantages.

Source timestamps

The advantage of source timestamps is that the original samples are used. The disadvantage is that if you have multiple source signals with different time bases, the sample points of the generated non-equidistant signals will have different timestamps. If you subsequently perform calculations based on these non-equidistant signals, they will be converted to equidistant signals and the calculations will be performed on equidistant signals.

Trigger timestamps

The disadvantage of trigger timestamps is that the original sample points are moved. The advantage is that all non-equidistant signals generated by the same profile have the same timestamps irrespective of their original time bases. Calculations are performed on the non-equidistant signals, and as result, a non-equidistant signal is given.

5.6 "Time, DB/cloud" profile

Use this profile if you want to write the measured data in supported databases by means of a DB/cloud time-based data store.

Profiles

Profiles

- As is
- Zeit_100ms
- Time, non-equidistant 2
- Time, DB/Cloud 3**

Add

Profile properties

Type: Time, DB/Cloud

Name: Time, DB/Cloud 3

Filtering :

☒ None ☐ Min

☐ Average ☐ Max

ID column value: Signal name

ID column size: 16 characters

Value column name: Signal ID

Value column size: 128 characters

☐ Add column for local time

Preview

I_TIME	I_ID	A_0_0	A_0_1	A_0_2
1:56:41.000 PM	Signal 0 name	3.9	NULL	NULL
1:56:41.000 PM	Signal 1 name	NULL	text 203	NULL
1:56:41.000 PM	Signal 2 name	NULL	NULL	35811
1:56:42.000 PM	Signal 0 name	1.1	NULL	NULL
1:56:42.000 PM	Signal 1 name	NULL	text 563	NULL
1:56:42.000 PM	Signal 2 name	NULL	NULL	8938

Aufzeichnung 1 : No signals linked
 ibaHD-Datenaufzeichnung 1 : No signals linked
 ibaHD längenbasierte Datenaufzeichnung 1 : No signals linked
 Kafka-Cluster zeitbasierte Datenaufzeichnung 1 : No signals linked

Profile properties

Type

Display of the type with the appropriate types of data storage (symbols)

Name

Enter an unambiguous and understandable name here.

Filtering

Select the signal value to be stored with reference to the time base set above.

- None: The actual value will be stored
- Average: The arithmetic average of all measured values, which were acquired in the interval (according to the time base, such as 10 ms or 100 ms) is formed and saved.
- Min: Only the smallest value of all samples measured within the defined interval will be stored.
- Max: Only the largest measured value of all measured values acquired within the defined interval will be stored.

Value of ID column, size of ID column

A table can also have an ID column, which can be used as the key. This column is denominated ID in *ibaPDA*. You can set whether you want to use such a column or not. If you want to use such a column, you can configure which value should be added. You can choose from the following:

- None: No ID column is used
- Fixed: A fixed text is written in the ID column. This can, for example, be used as identification of the *ibaPDA* system if several *ibaPDA* systems write in one table.
- Text signal: The value of a text signal
- Signal ID: The ID of the signal to which this profile is assigned, e.g. [10:2]
- Signal name: The name of the signal to which this profile is assigned
- Signal comment 1: The first comment of the signal to which this profile is assigned
- Signal comment 2: The second comment of the signal to which this profile is assigned

The ID column is created with the data type NVARCHAR(x), whereby x is the size that was set in the field *Size of the ID column*.

Name of the value column, size of the value column

Use this option to determine the values column in which the value of each signal assigned to this profile is written. Available for selection are:

- Fixed: A fixed column name. This means that the data of each signal is written in a separate line. This also means that you should use an ID column to be able to determine the signal to which the data belongs.
- Signal ID: The column name is derived from the signal ID.
- Signal name: The column name is derived from the signal name.
- Signal comment 1: The column name is derived from the first comment of the signal.
- Signal comment 2: The column name is derived from the second comment of the signal.

If you do not choose “fixed,” then a line will receive the data from several signals. The notation is more efficient, but not mandatory.

If you want to write text signals in the table, you must specify the size in number of characters.

Add column for local time

The default time column I_TIME shows the UTC time. If you enable this option a column showing the local time will be added.

Preview

The preview shows which columns will be created based on the ID and value columns settings.

Other documentation



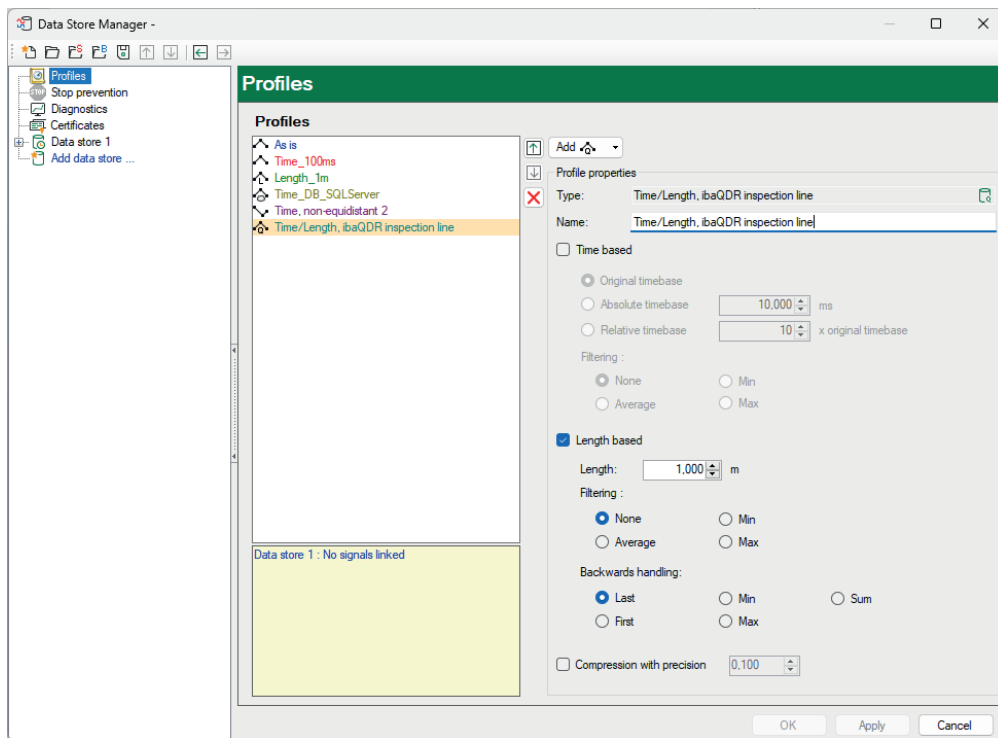
You will find more information about DB/cloud data storage in the respective manual for the product, such as *ibaPDA-Data-Store-SQL-Server-xxxx*.

5.7 "Time/length, ibaQDR inspection line" profile

With this profile, both time and length-based data is recorded in data files. Since this profile type can only be used with *ibaQDR* and when using an *ibaQDR data store* or *ibaQDR inspection line data store*, it is also called a QDR profile.

When adding an *ibaQDR data store*, two base profiles are automatically created. When using *ibaQDR*, you may add further QDR profiles if required.

The QDR profiles are not available for signal selection for other types of data storage.



Profile properties

Type

Display of the type with the appropriate types of data storage (symbols)

Name

Enter an unambiguous and understandable name here.

Time-based, length-based

By enabling one of these options or both of them, you determine how the data is written in the data file.

With the time-based recording, the settings are the same as for the "Time" profile.

With the length-based recording, the settings correspond. A length interval can be set here instead of an absolute time interval. The standard deviation is 1 m. You can enter both larger and smaller values.

Other documentation



The profile *Time/length* is only suitable for *ibaQDR data stores* or *ibaQDR inspection line stores*. You will find more information about this in the manual for *ibaQDR*.

Note



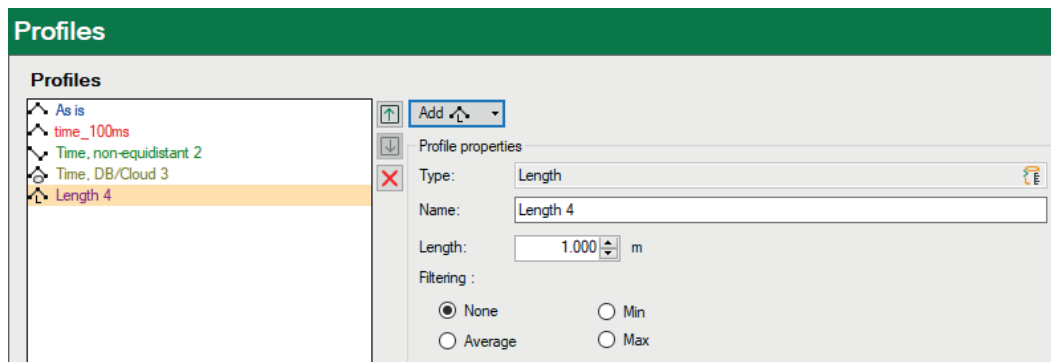
Values are always stored in REAL format as soon as a storage profile is assigned to the signals (except "As is"), even if the original samples are of INT16 format (e.g. from an ibaPADU)!

Exception:

32 bit integer values are not converted to floating point values when written to a data file.

5.8 "Length" profile

This profile type is used to record measured values in a length-based ibaHD data storage. It determines at what intervals (meters) measured values are written in an ibaHD store. The conversion of measured values to a length base is configured in the definition of the ibaHD data storage.



Profile properties

Type

Display of the profile type with the appropriate types of data store (symbols)

Name

Enter an unambiguous and understandable name here.

Length

Adjust the length of the section interval here, at which a measured value should be recorded. The default setting is 1 m.

Filter

Select which value of a signal, related to the set distance interval, is to be stored in the record.

- None: The actual value will be stored
- Average: The arithmetic average of all measured values, which were acquired in the interval (according to the time base, such as 1 m or 10 m) is formed and saved.
- Min: Only the smallest value of all samples measured within the defined interval will be stored.
- Max: Only the largest measured value of all measured values acquired within the defined interval will be stored.

Other documentation

For more information about ibaHD data storage, please refer to the *ibaHD-Server* manual.

5.9 “Time, ibaHD” profile

This profile is dedicated for timebased recording in ibaHD-Server. The profile offers 3 different modes, which support the following types of recording:

- As is
- Equidistant
- Non equidistant

Furthermore, the profile allows to adapt the aggregation factor.

Profile properties**Type**

Shows the type with the appropriate data storages (symbols)

Name

Enter an unambiguous and understandable name here.

Mode

Select the the desired recording mode.

- As is
The signals are stored as they are. This means, equidistant signals will be stored equidistantly, non equidistant signals will be stored non equidistantly. There are no further settings except the aggregation mode.

■ Equidistant

The signals are resampled. The new timebase can be configured as an absolute timebase or an integer multiple of the original timebase. Filtering can be used to determine which value will be stored. This profile has the same options as the standard "Time" profile.

For more information about this recording mode see ➤ *"Time" profile, page 16*

■ Non equidistant

A sample is taken and stored each time the value of the trigger signal changes. In the case of a digital trigger signal a sample will be taken on both rising and falling edges. The timestamp of the sample can be taken from the source signal or from the trigger signal. This is only relevant if the time bases of source and trigger signal are different. This profile has the same options as the standard "Time, non-equidistant" profile.

For more information about this recording mode see ➤ *"Time, not equidistant" profile, page 17*

Aggregation mode

For all 3 modes the profile offers the option to set the timebase for the different aggregation levels automatically or manually.

If you want to set the timebase manually, select *Custom aggregation timebases* in the drop down list of this field.

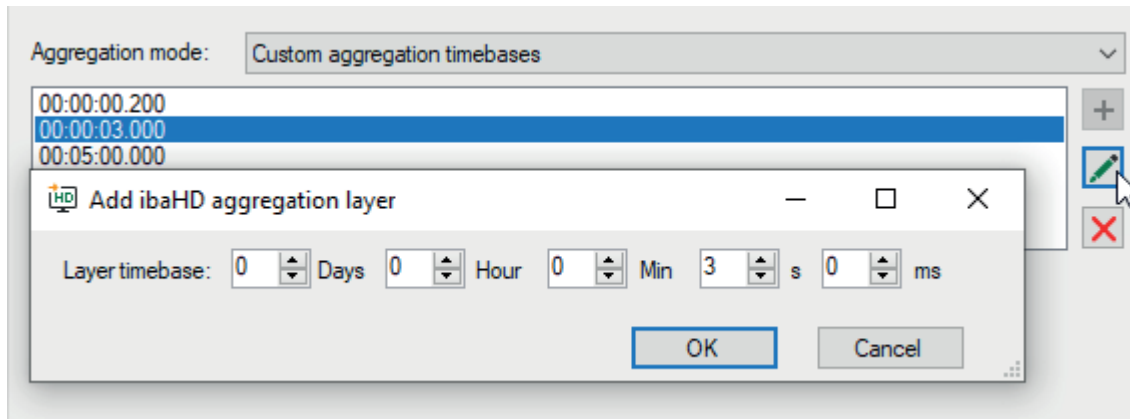
The current timebases will then be displayed.

On the right side you will find a number of buttons with the following functions:

	Add new timebase; only available if there are less than 5 timebases.
	Edit selected timebase
	Remove selected timebase

If you want to change a timebase, select it in this list and click on the <Edit> button.

You then can change the value in the following setup dialog.



Other documentation



For more information about the aggregation levels, please refer to the *ibaHD-Server* product manual.

6 Configuring data storage

The term *data storage* describes a method for storing measured data in data files or other media depending on particular trigger events or other controls. You may define and configure any number of data storages, which are capable of recording data independently from each other.

Note



The following chapters explain the key configuration steps using the example of the standard data storage (time-based in data files). Many of these descriptions also apply accordingly for other types of data storage.

Configuration steps deviating from this and settings for other types of data storage, such as *ibaQDR*, *ibaHD*, DB/Cloud, etc. can be found in the respective manuals for the software products.

The number of independent data storages is limited by the license key (dongle). A *ibaPDA* standard package contains two data storages in data files, which can be used at the same time. With additional licenses, upgrades in steps of two additional data storages are available. Each data storage creates its own data files using its own recording parameters such as storage profiles, trigger settings and hard disk cleanup strategy. All data storages can work simultaneously and create data files. Moreover, each data storage can create an overlapping recording, consisting of two or more files. The available data storages are listed in the tree of the data storage dialog, each with its sub-branches *Trigger mode*, *Signal selection* and *Files*.

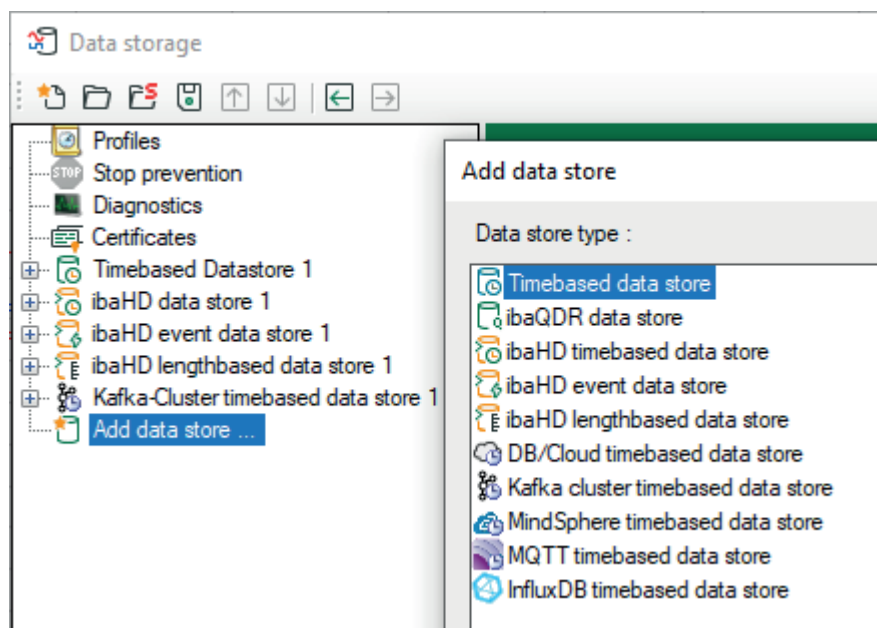
Note



The following restrictions apply to data storages in data files:

- Module numbers: 0 to 1048575
 - Highest signal number: 524287
 - Highest sub-signal number: 16383 (e.g. for bit decoder)
-

6.1 Add a data storage



There is automatically a standard data storage (time-based) available, but no signals are assigned to it. You may rename and configure this default data storage. In order to add a new data storage, click on the blue link *Add data storage* in the tree structure. You can also right-click on the data storage name in the tree and choose *Add data store* from the context menu.

If you do not assign a storage profile to the signals to be recorded, they will be assigned to the profile "As is". In this case, the signals are recorded with the original time base according to the General settings or the Module settings.

6.2 Basic settings for data storage

 The image shows the configuration window for 'Timebased Datastore 1'. The window has a green title bar. Inside, there are several settings:

- ☐ Locked
- ☒ Active
- Name:
- File password: ...
- ☐ Flush recorded data to disk every s
- ☐ Use asynchronous file access
- ☒ Process signals in parallel
- Data store index:

Locked

If you enable this option, then a data storage configuration can only be changed or disabled if the current user has the corresponding rights with the user management enabled.

Active

A data store must be enabled in order to work. The advantage of configuration and activation being separated is that you can define and configure several data storages without being obliged to use them all at the same time when you disable the storages you do not require. Just disable the data storages you do not want to use.

To enable a data storage, click on the major branch of the relevant data storage and check the *Active* option.

Data store index

This index is automatically assigned by the system and is only shown here. The index is increased by 1 every time a data store of the same type is added. The index is important if the diagnostic functions `DataStoreInfo...()` are to be used. For each data storage type there is a corresponding function (`DataStoreInfoDB`, `DataStoreInfoHD` etc.) In this function the index must be entered as an operand in each case in order to receive the diagnostic data from the correct data store.

Data store name

Enter here a meaningful name for the data storage here.

File password

You can protect the data file from being opened without authorization by using a password. The password must be at least 6 characters long.

To open a protected data file with *ibaAnalyzer* later, the user must know and enter the assigned password here. If no password has been set yet or if you want to change the password, click the <...> button right next to the field.

The file password is also used by the playback interface and the offline trend graph (*ibaQPanel*) as well as by *ibaDatCoordinator* (post-processing) and *ibaHD server* (import), provided it has been configured accordingly there.

Flush recorded data to disk every ... s

ibaPDA writes measured data automatically into the data file while making optimal use of the integrated data compression.

Enabling this option prompts the system to write the data to the data file regularly after an adjustable time period has elapsed. This is necessary, for example, if data files will be analyzed with *ibaAnalyzer* while they are still being written. If this option is enabled, however, the data files will become larger, since data compression can no longer be used optimally.

Use asynchronous file access

By default, this option is disabled. This means that all write operations for data storage and also data acquisition are processed sequentially in one thread.

Especially when using external drives for data recording (NAS, USB disks), where access is significantly slower than with an internal, local hard disk, data acquisition may be slower and the interrupt buffer will fill up. In the worst case, the buffer will become full and data acquisition will stop. This can also occur when writing to different partitions of the same physical drive, which is still being used by other processes.

Another disadvantage is the failure of all data stores if one drive is not available, despite the fact that different drives are being used.

Enabling this option decouples data acquisition and data storage, i.e., data acquisition and data storage are then processed in different threads. This prevents data acquisition from being negatively affected by slow file access times during data storage.

When using multiple physical drives (NAS, USB) for different data stores, if one drive is ever unavailable, only the corresponding store is affected. The other stores can continue writing.

The disadvantage of this operating mode is a higher memory requirement. The buffer memory for write commands to be executed is 500 MB per data store. If this limit is reached, then the recording stops and restarts after 5 s.

Process signals in parallel

This option is especially advantageous if you want to process and store a large number of signals. Parallel processing significantly increases performance.

6.3 Trigger mode

In order to open the trigger mode dialog, click on the "Trigger mode" subbranch of the relevant data storage in the tree.

This dialog contains a number of settings relevant for data storage control. Different trigger modes for starting and stopping a data storage are available. You can specify parameters such as pre- and post-trigger time and the behavior of the data storage in specific situations.

All settings remain until the next change, even if the acquisition is stopped or the program is closed. Any change in the settings is saved to the current data storage configuration file at the time you press the <OK> button and exit the dialog.

Note



The trigger itself does not affect the measurement. It only determines when and how the data is saved.

Triggers only work when the measurement is running.

6.3.1 Maximum file time

This setting specifies the time limit of a data file. No data file of the corresponding data storage will exceed this time limit.

The limitation also applies to triggered recordings. This is particularly useful for avoiding "over-long" data files in case a defined stop trigger does not fire, e. g. due to incorrectly set levels or edges. The time limit is dominant over the stop trigger. If you want to use a stop trigger for stopping the recording, make sure that the time limit is high enough for the expected recording time.

If neither a start nor a stop trigger is set, the recording is only time-controlled. This is a continuous recording with a number of consecutive data files of the same length, i.e. recording time.

You may directly enter the values for Hour, Minute and Second or use the arrow buttons.

The maximum recording time is one week (168 h : 59 min : 59.9 s).

Please note, that the data file size can grow significantly depending on number of signals, profile and file time. The data file size is limited to 64 GB.

6.3.2 Start trigger

Trigger modes

You can choose between several trigger modes when using a start trigger for data storage:

No start trigger

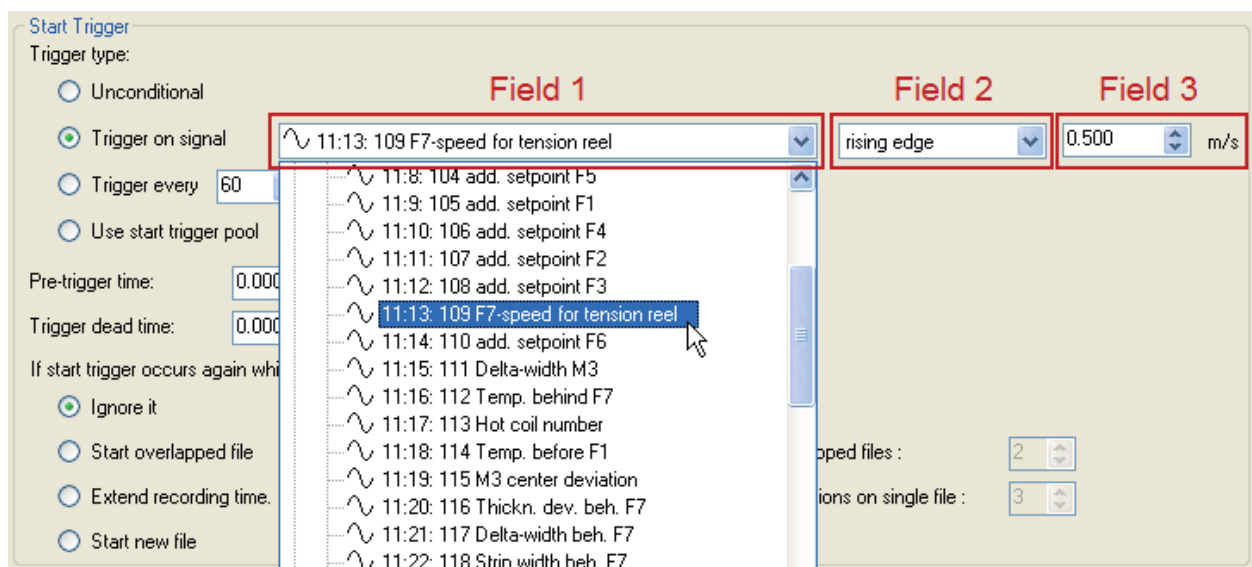
If this item is checked, no trigger is enabled. In this case, the recording will start immediately at the start of the measurement or when pressing the "Go" button.

Trigger on signal

If you want the trigger to fire on a measured signal or a virtual signal, you need to check "Trigger on signal" in the trigger mode dialog.

- Field 1 Dropdown list for signal selection
(available analog and digital signals)
- Field 2 Dropdown list for selecting edges or levels
- Field 3 Dropdown list for selecting the trigger level value
given in the specific physical unit
(field 3 is only enabled in case of analog trigger signals)

Both analog and digital signals can serve as triggers. The signal to trigger on is to be selected from the drop-down lists in the fields nearby (see picture below, field 1). In the drop-down list, you will find the well-known signal tree containing available signals. Select the signal you want to use as trigger signal.



Depending on whether a digital or an analog signal was selected, the fields 2 or 3, respectively, are offered allowing the trigger event to be defined more specifically.

As for analog signals, you can choose between level or edge triggers including a predefined level (field 3). An edge in the context of analog values is a transit of a measured value through a predefined level.

- Rising edge: The value exceeds the level.
- Falling edge: The value undercuts the level.

The setting *above/below level* is available for start trigger only. It is used to make sure that the recording will start immediately with start of acquisition if the start trigger condition (value is above or below level) is fulfilled at this time. See also explanation in the tip below.

Start Trigger

Trigger type:

- ☐ Unconditional
- ☒ Trigger on signal
- ☐ Trigger every 60 minutes starting at 00:00
- ☐ Use start trigger pool

Field 1: 11:13: 109 F7-speed for tension reel

Field 2: above level (dropdown menu open showing: rising edge, falling edge, above level, below level)

Field 3: 0.500 m/s

Pre-trigger time: 0.000 s ☐ Keep pre-trigger data in memory

Trigger dead time: 0.000 s ☐ Trigger at the start of the acquisition

As for digital signals, you can choose between level or edge triggers including the 2 levels logical 0 (FALSE) and logical 1 (TRUE).

Start Trigger

Trigger type:

- ☐ Unconditional
- ☒ Trigger on signal
- ☐ Trigger every 60 minutes starting at 00:00
- ☐ Use start trigger pool

Field 1: 11:11: F1 Stand loaded

Field 2: rising edge (dropdown menu open showing: rising edge, falling edge, Logical 1, Logical 0)

Pre-trigger time: 0.000 s ☐ Keep pre-trigger data in memory

Trigger dead time: 0.000 s ☐ Trigger at the start of the acquisition

Information on the use of trigger edges

Tip

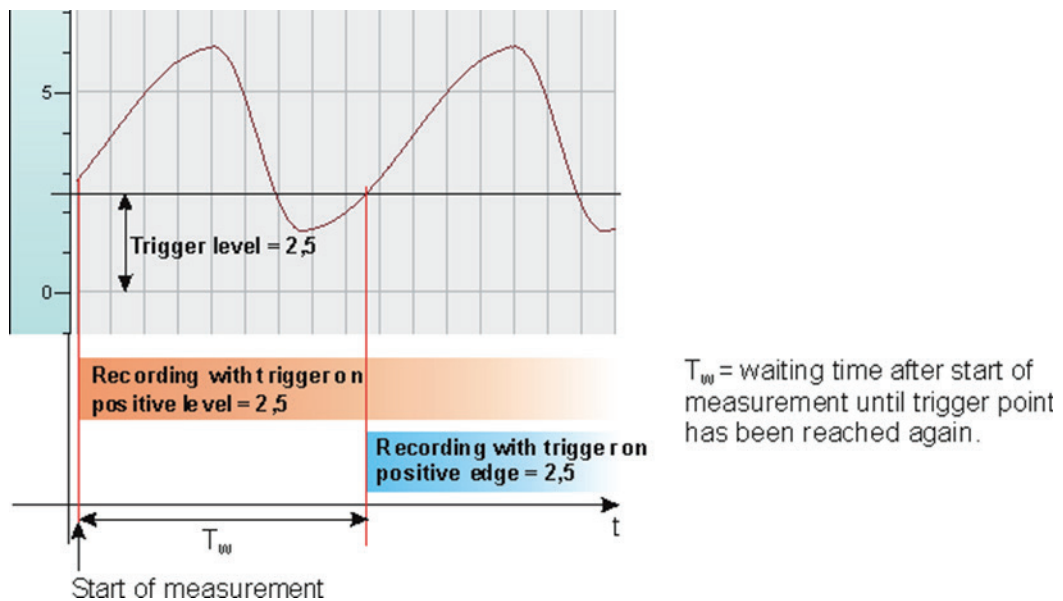


Care must be taken when using trigger edges!

In order for a trigger to fire at a positive or negative edge, the value specified in field 3 must be at least slightly exceeded or fallen below. Otherwise, no trigger will occur.

Therefore, it is often more useful to define a level trigger, which fires as soon as the signal reaches or exceeds/falls below the level defined in field 3. If this is the case already at the time the measuring starts, the recording will start immediately.

The diagram shows the differences between the recording starts, which might occur depending on the trigger conditions and the time the measurement starts.



Periodic trigger

If you want to use a start trigger always at a certain time on a periodical basis, you can check the "Trigger every ... minutes starting at ..." option.

This is useful if you want records to be started every 8 hours, e. g. based on a work shift.

Enter the period given in minutes, or select it from the input field. Value range is from 0 to 1440, which equals one day. Then enter or select the start time for the first trigger. Value range is from 00:00 to 23:59, which equals one day.

Example

"Trigger every 480 minutes starting at 06:00" will generate the first start trigger at 06:00, followed by further start triggers every 8 hours (= 480 min), that is at 14:00 and 22:00. The last trigger on that day will fire at 22:00. The next trigger will then fire at 06:00 on the following day.

In order to ensure the recording of data files of exactly 8 hours without a sample lost, set the "Maximum file time" to 9 hours and enable the "Start new file" option (in case another start trigger fires while the recording is already running).

If you want the recording to start immediately at acquisition start or as soon as you apply a new data storage configuration, you also need to check the "Trigger on start of the acquisition" option. The first data file created will then contain the range from the restart of the acquisition or data storage to the next trigger.

Tip

The periodic trigger at a specified start time is useful for creating data files beginning at second 0 of one minute. If you, for instance, want to create data files of 1 minute length from second 0 to 59, use the periodic trigger with the settings "Trigger every 1 minute(s) starting at 12:34" (any time).

Set the "Maximum file time" to 2 minutes and enable the "Start new file" option (in case another start trigger fires while the recording is already running).

Use start trigger pool

Enable this option if you want to use the trigger signals from the start trigger pool. See [➤ Trigger pool, page 38](#).

Pre-trigger time

Setting a possible pre-trigger time.

If this function is selected by entering a time value, recording begins immediately when the measurement starts. In this case, the measured data is stored periodically in a file, with its length of time corresponding to the pre-trigger time, or in the computer memory. If the trigger condition is met, the incoming data is added to the data buffered during the pre-trigger time. Therefore, the file start does not match the trigger time at the end of recording.

In the offline analysis with ibaAnalyzer, the trigger times of the start and stop trigger are marked by two arrows in the graph.

If you want to use the start trigger pool, you can determine that the trigger name is added to the filename of the data file. Go to "Files" in the configuration dialog of the data storage and enable the "Add start trigger name" option.

Keep pre-trigger data in memory

Enable this option if you prefer to store the pre-trigger data in the computer memory instead of a file on the hard disk. Be sure to have sufficient RAM installed when you choose this option.

Trigger dead time

This property is available for every start trigger type (signal trigger, periodic trigger and start trigger pool). The trigger dead time determines the time of suppressing subsequent triggers after a trigger occurred.

If the dead time, for instance, is set to 5 seconds, all other triggers are ignored for the duration of 5 seconds after the first trigger occurrence.

Trigger at the start of the acquisition

If you have defined a start trigger (any type) and restart the acquisition or apply a new data storage configuration to the server, the recording will not start unless the trigger fires at the same moment. Without a trigger occurrence at this time, you will have to wait until the trigger fires and the recording starts.

Check this option if you want to make sure that the recording starts immediately after restart of acquisition and/or data storage regardless of any start trigger condition.

If start trigger occurs again while file is already recording, then:**Ignore it**

Selecting this option will cause the system to ignore any new start trigger during a running recording for as long as the stop trigger occurs.

Start overlapped file

This option should be used to save measured data in separate files, in case they depend on more than one independent but overlapping trigger events.

If a start trigger occurs, a file is created, in which the measured data will be saved. If another start trigger fires during a running recording, another file is created for storing the measured data. When the first stop trigger fires, the first file will be closed; the second file will be closed when the second stop trigger occurs.

So, this mode allows the start trigger of the subsequent recording to occur before the stop trigger of the recording currently running, i.e. temporarily parallel recordings in two successive files of the same file series. This mode is needed for overlapping, linked processes.

The overlapped mode is not limited to two files only. Three or more files are also possible, depending on the number of start triggers occurring before the first stop trigger fires and the defined maximum number of files.

Maximum number of overlapped files:

If you have selected the "Start overlapped files" option, you should set the number of permissible overlapped files. Usually, two are sufficient. You may, however, adjust the number according to your needs, i. e. how many start triggers you expect to occur before the stop trigger fires.

Rule of thumb: number of possible start triggers during a recording + 1 = max. number of overlapped files.

Upper limit is 10.

Extend recording time

As an alternative to the overlapping mode, choose this option if you prefer to extend the running recording in case another start trigger occurs during the running recording. Each additional start trigger extends the recording time by the time which is set as "Maximum file time" above. This occurs as often as set in the "Maximum number of extensions on single file" field. If the max. number of extensions is reached, all subsequent start triggers will be ignored. The recording will then be stopped after the last maximum recording time has elapsed. Of course, the recording is stopped immediately by any stop trigger.

Maximum number of extensions on single file

If you have selected the extended recording mode, you may set the maximum number of extensions in this field. If this limit of extensions is reached, the recording is stopped after the maximum file time at the latest.

Start new file

The current data file is closed and a new file created. So, the start trigger can serve as a stop trigger at the same time.

6.3.3 Stop trigger

The settings for the stop trigger are made in the same way as those for the start trigger. Here, both analog and digital signals can also be used as triggers.

No stop trigger

A running recording is only stopped by stopping the measurement or after the maximum file time has elapsed.

Trigger on signal

See explanation for start trigger above.

Trigger on reception of text signal

Instead of using a measured signal as stop trigger, you can also stop the recording by the reception of a text signal. In this case, the stop trigger is fired by the reception of the text signal selected in the field nearby. The contents of the text signal are not relevant for this function.

Please note the relation between this function and the naming of the data file in correspondence with the text signal in the "Files" branch of the data storage configuration ("Files - File name - Add text signal").

Use stop trigger pool

Enable this option if you want to use the trigger signals in the stop trigger pool. See [↗ Trigger pool, page 38](#).

Post-trigger time

Setting an optional post-trigger time.

If this function is selected by setting a time, recording does not stop until this post-trigger time has expired. Exceptions: if the stop trigger does not occur, the "Maximum recording time" setting will automatically stop the recording. If the stop trigger fires only shortly before the maximum recording time ends so that there is not sufficient (memory) space left for the full post-trigger time, recording is nevertheless stopped after this time, i.e. the post-trigger time is partially cut off.

In the offline analysis with *ibaAnalyzer*, the trigger times of the start and stop trigger are marked by two arrows in the graph.

6.3.4 Trigger pool

With the trigger pool, *ibaPDA* provides an easy way to configure multiple trigger signals as start or stop triggers for data storage.

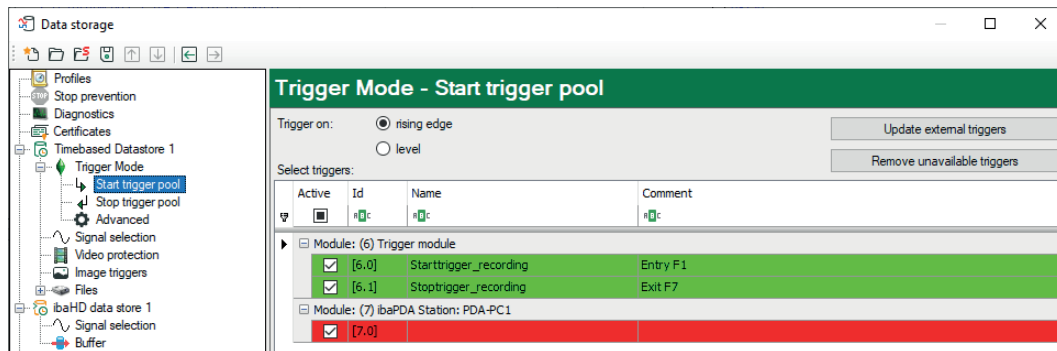
The signals of the trigger modules can be used as triggers for the recording.

For more information on trigger modules, please refer to part 3 *Trigger module*.

If several signals should be used as start or stop triggers for recordings, the setup of start trigger pools and stop trigger pools is recommended.

Each data storage has a start trigger pool and a stop trigger pool. A trigger pool is basically a list of signals taken from all trigger modules. If one of the triggers fires then the entire trigger pool is triggered as well. The trigger can fire on a rising edge of a trigger signal or at each sample with the signal = 1 (TRUE).

You will find the configuration dialog for start and stop trigger pools under the "Trigger mode" branch in the tree structure of each data storage.



Use the drop-down menu in the first column to select the trigger signals belonging to the trigger pool.

The selected trigger signals are highlighted in green color.

A red-highlighted row indicates that the trigger signal is no longer available. This is the case, e. g., if you have disabled a trigger module. Missing signals are ignored. You can remove them from the list by clicking the <Remove unavailable triggers> button.

The first row of the trigger grid can be used to filter and sort the trigger signals, just as in a spreadsheet. This can be very helpful if you want to find one particular signal in a long list of numerous trigger signals.

Clicking on the "ID", "Name" or "Comment" column headers will sort all trigger signals by the criterion selected in ascending or descending order.

If you click on the small filter icon in the upper right corner of the "ID", "Name" or "Comment" column headers (only visible by hovering the mouse), you can select one of the entries for filtering. You can also enter an ID, a name or a comment, or parts of it, in the corresponding cells.

Example:

If you know that the name of the trigger signal you are looking for starts with “Trigger...”, just start typing the name in the respective cell and all the corresponding triggers will be listed below.

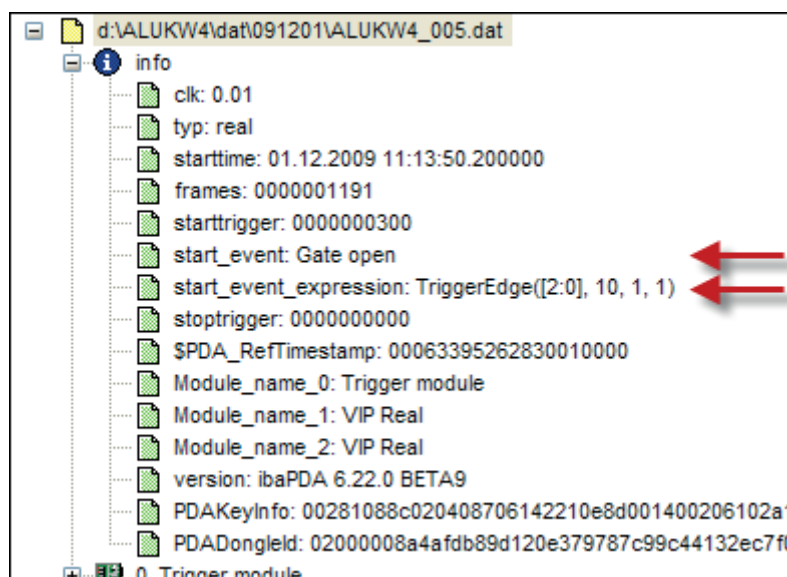
Name	Comment
Trigg	
Trigger module	
Trigger final speed	Triggers if final speed has been reached
Trigger rollforce	Triggers if rollforce exceeds min value
Trigger Speed acceleration	Triggers if acceleration
Trigger speed min	Triggers if speed is above min speed

The keys <Ctrl> or <Shift> allow you to enable/disable multiple signals at the same time.

If you click the “Active” column header then the value (active/inactive) of the selected row is copied to all the rows underneath it. Double-clicking on the “Trigger module” header enables/disables all the trigger signals of this module.

If you want to use a start trigger pool or a stop trigger pool as trigger, you should enable the option “Use start trigger pool” or “Use stop trigger pool”, respectively, in the trigger mode dialog.

When using a trigger pool, recording can be started or stopped by any desired trigger within the trigger pool. The name of the triggering trigger is stored in the data file in the same way as the info fields “start_event” for the start trigger and “stop_event” for the stop trigger. The expression following the trigger is also stored in the info field “start_event_expression” or “stop_event_expression”, respectively. The name of the start trigger can also be added to the filename.



The <Update external triggers> button is only relevant in conjunction with the multistation mode where multiple *ibaPDA* systems exchange signals with one another, synchronized or un-

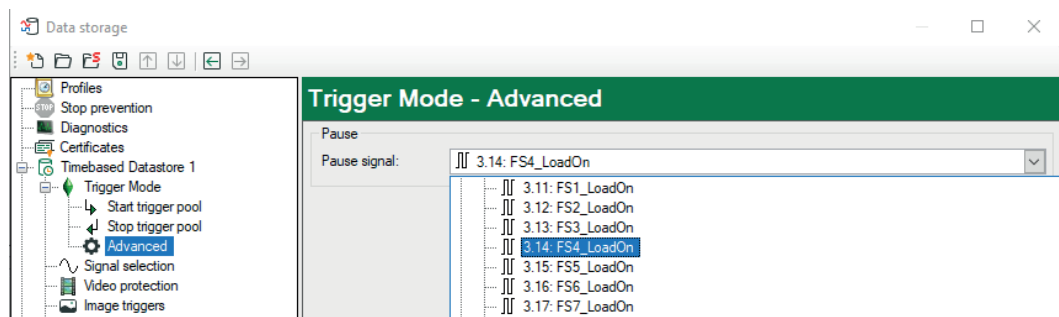
synchronized. Because for non-synchronized operation the trigger signals from other stations (external triggers) are not automatically updated, you have to click on this button in order to get the current trigger status from the other stations.

6.3.5 Advanced

Pause

Data storage can be paused by a digital signal. During long-term recordings, it might be useful to pause the data storage yet without closing the data file. For example, this could be the case in shut-down times of machines during a work shift when no measured data arise, but the data file is intended to cover the entire shift.

While a data storage is paused, all measured values in the data file are replaced with invalid values, which will be displayed as a gap in ibaAnalyzer. Signals with INT, WORD and BYTE data types are saved to the data file as 16 bit integer values. Since there is no invalid bit for these data types in the data file, these values will be replaced with 0.



The period of the ibaPDA server acquisition loop is equal to the smallest common multiple of all module time bases, but no less than 50 ms.

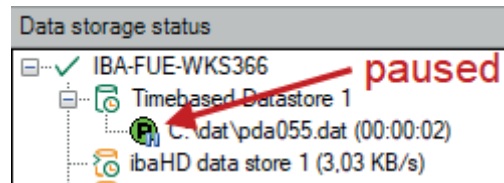
The pause signal is evaluated only once per acquisition loop. The signal is 1 (TRUE) if it remains 1 (TRUE) for the whole duration of the acquisition loop period. It is 0 (FALSE) as soon as 1 sample is 0 (FALSE) during the period of the acquisition loop.

This means that the data storage will be paused a certain time x after the rising edge of the pause signal. The time x is between 0 and the acquisition loop period.

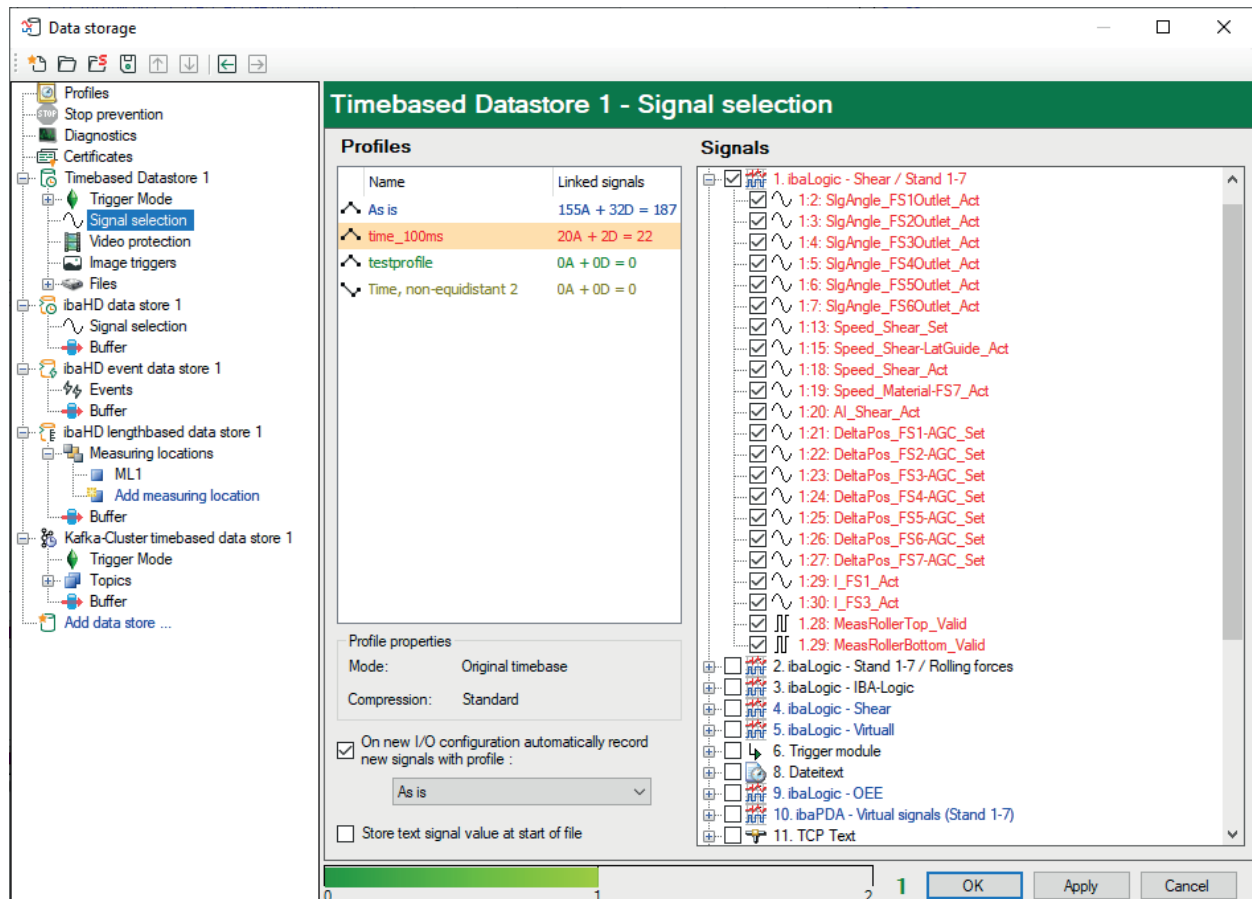
This also means that the data storage will be resumed a certain time x before the falling edge of the pause signal.



The data storage status tree shows a pause icon when the recording is paused.

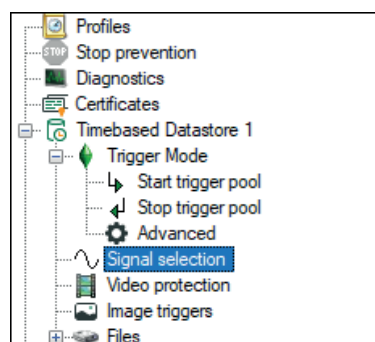


6.4 Signal selection



Signals must be assigned to a data storage profile in order to be recorded in a recording profile. A simple way is to assign all signals to be recorded to the "As is" standard profile. However, this could result in very large data files, depending on the number, sample rate and maximum recording time limit.

In order to select signals for recording, mark the *Signal selection* sub-branch in the tree of the data storage dialog.



In the right part of the dialog window, the available recording profiles are listed. Next to it on the right is the signal tree with all available, i.e. enabled, signals. Only the signals shown here can be assigned to a recording profile.

Note



A signal can only be assigned to one profile per data store.

If a signal will be recorded with different profiles, different data recordings must be defined for this purpose or copies of the signal must be generated via virtual modules.

You may, however, use as many profiles as you like in a data store.

On new I/O configuration, automatically record new signals with profile:

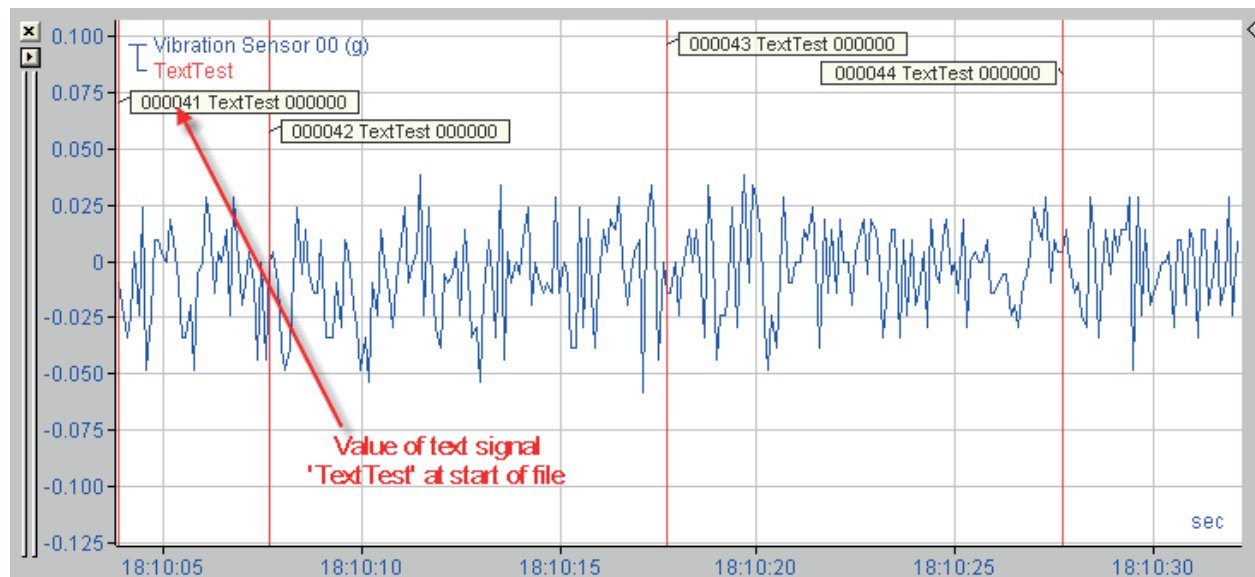
In the lower part of the dialog, you can choose a profile from a pick-list which will be used as default profile for new signals. If you add new signals to the I/O configuration, these signals will be assigned automatically to this default profile and recorded.

Store text signal value at start of file

Enable this option if you want to ensure that the value of an assigned text signal is always saved already at the start of each measurement file. The value of the text signal will then be available right from the beginning of the measurement file for subsequent evaluation.

If you do not enable this option, the values of the text signals are only saved in the measurement file then, i.e. when the value changes. Therefore, if the value of a text signal does not change during the recording, no text appears in the file. If the value changes during recording, the value of the text signal would only be recorded from that point on.

The following figure shows the result with this option enabled in *ibaAnalyzer*.

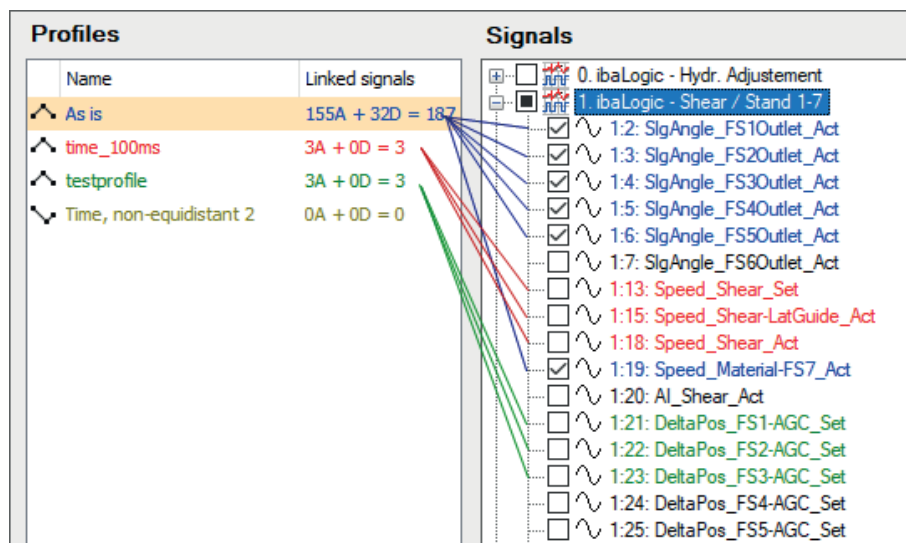


6.4.1 Assigning a signal to a storage profile

A tick in the selection box beside a signal shows that this signal is assigned to a profile marked in the left section of the dialog window. In order to assign a signal, the profile must be selected first and then the signal. If you want to assign all signals of a module to a profile at once, the relevant module needs to be checked in the selection box of the module. All signals of this module will be checked, too.

Signals not selected are either assigned to a different profile or not assigned at all. Whether a signal is already assigned to a profile can be checked by clicking on each profile and checking the selection box of the relevant signal. If the signal is not checked, it is not assigned.

More convenient is the color check: the colors of the signal names correspond to the colors of the profiles they are assigned to. If you extend a module in the tree, you see at once which signal is assigned to which profile. A black signal name indicates that the signal has not been assigned to a profile yet.



6.4.2 Checking the profile assignment

If you like to check quickly whether a storage profile is assigned to a signal, select the profile in the profile list.

Below this list, you will find information about the number of analog and digital signals assigned to this profile.

At the same time, you will see checkmarks at the modules, whose signals are assigned to this profile, in the signal tree right to the profile list.

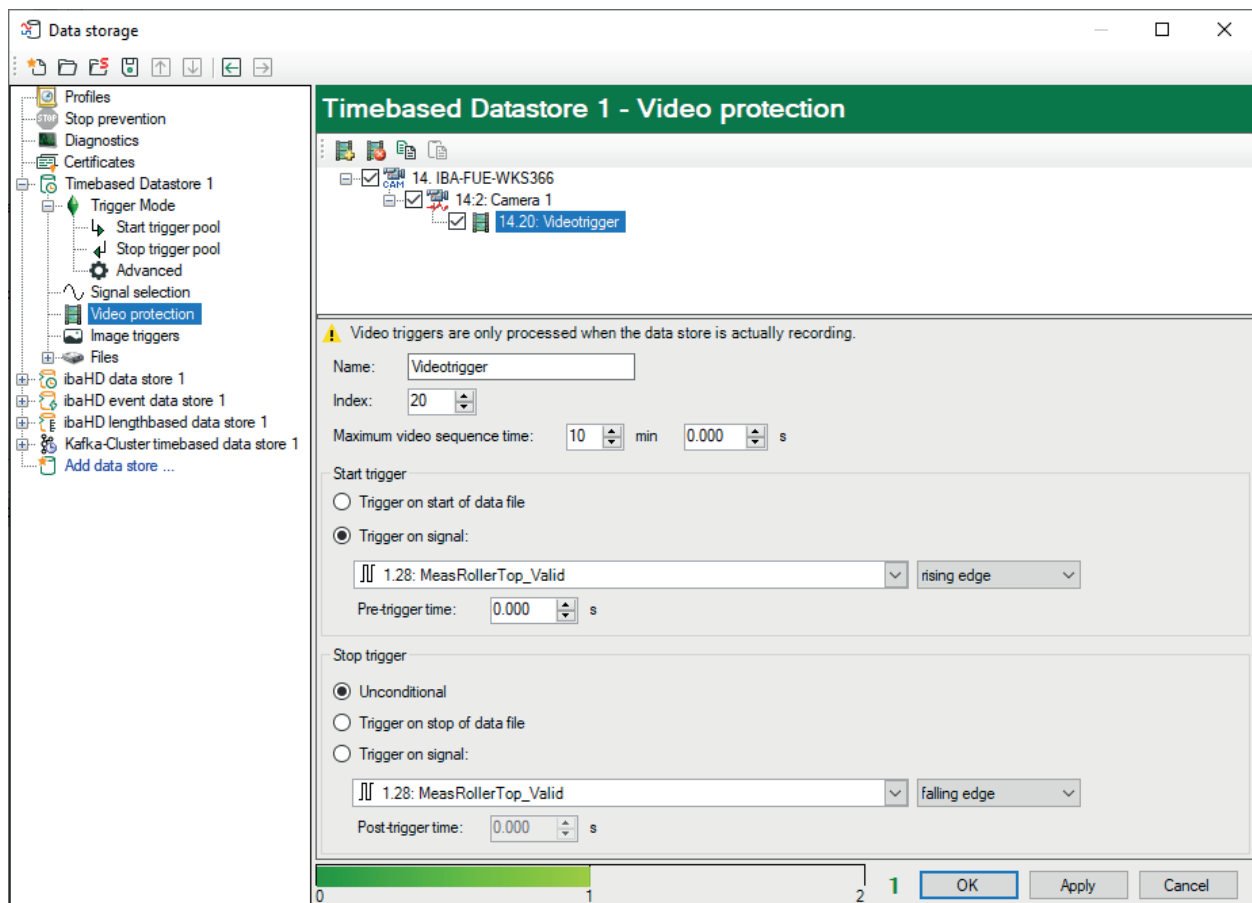
A gray square within the selection box means that only a part of the signals of the corresponding module is assigned to this profile.

A checkmark on a white selection box indicates that all signals of the corresponding module are assigned to this profile.

6.4.3 Disabling the profile assignment

In order to remove a profile assignment, just uncheck the corresponding signal or module.

6.5 Video triggers



All *ibaCapture* modules configured in the I/O manager, i. e. all *ibaCapture* servers and their associated cameras, are listed in this tree.

For each camera you can add, delete and configure video triggers for a reliable video recording. Every video trigger consists of a start trigger and an optional stop trigger. The start and stop trigger can be any measured signal, virtual signal or trigger module signal.

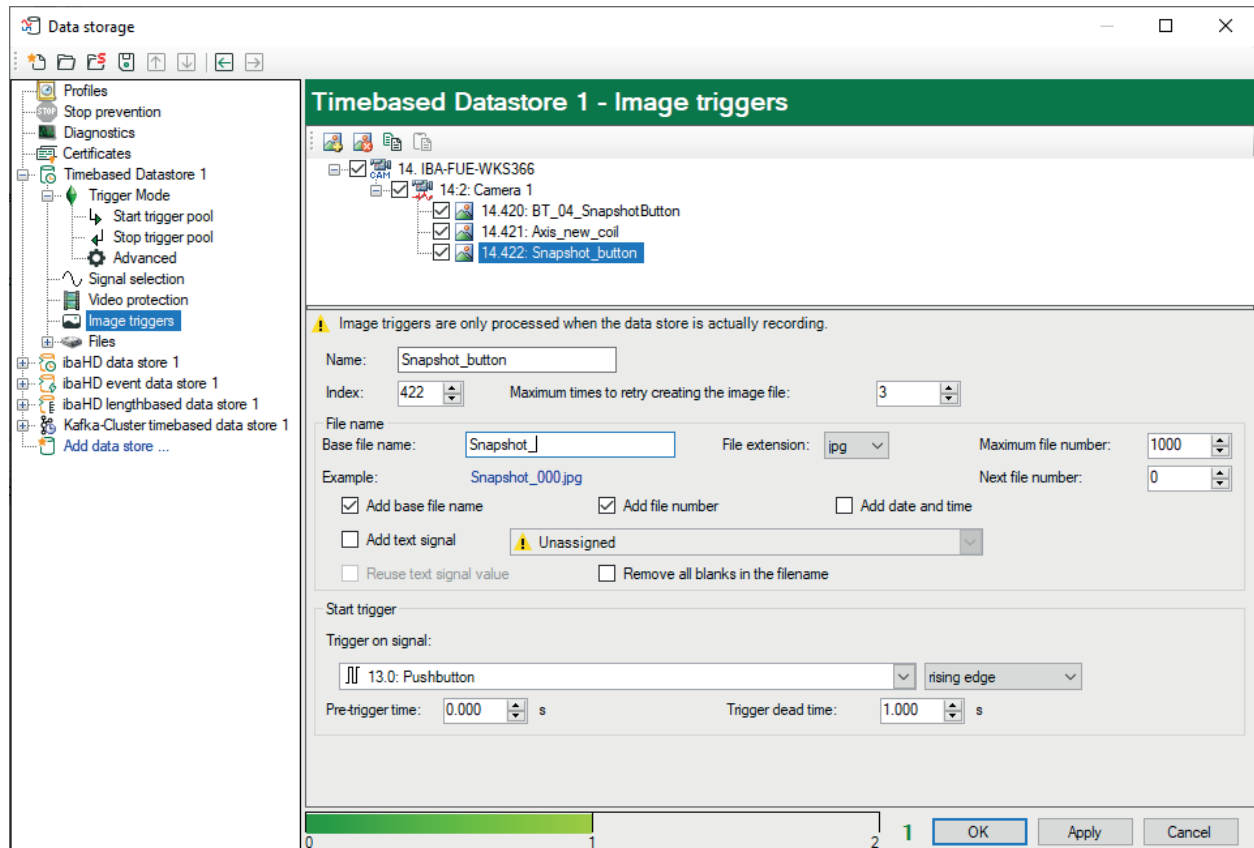
Other documentation



For a detailed description of video triggers and their configuration, please refer to the corresponding *ibaCapture* manual.

6.6 Image trigger

With image triggers you can store event-controlled snapshots or stills from video recordings as an image file.



All *ibaCapture* modules configured in the I/O manager, i. e. all *ibaCapture* servers and their associated cameras, are listed in this tree.

For each camera, you can add, delete and configure image triggers.

Every image trigger is given a name and index. The file name and type of the image file to be generated can be configured individually for each image trigger. You define the trigger for the snapshot with the start trigger. You can use any measured signal, virtual signal or trigger module signal as the start trigger.

There is a default path configured as base directory for the image files:

`c:\iba\ImageData`

Sub-directories are automatically created for *ibaCapture*-Server, cameras and image triggers.

Other documentation



For a detailed description of image triggers and their configuration, please refer to the corresponding *ibaCapture* manual.

6.7 Data file information

Data storage

Timebased Datastore 1 - Files

File name

Base file name: Maximum file number:

Example: Next file number:

☒ Add base file name
 ☒ Add file number
 ☒ Add start date and time
 ☐ Add start trigger name

☐ Add text signal
 ☐ Reuse text signal value
 ☐ Remove all blanks in the filename

☐ at file open / start trigger
 ☐ at file close / stop trigger

File location

Base directory:

User name:

Password:

Backup directory:

Subdirectories

Example:

☒ None
 ☐ Day based
 ☐ Week based
 ☐ Hour based
 ☐ Month based

☐ Use multiple levels of subdirectories
 ☐ Restart file number when subdirectory changes

Maximum subdirectories:
 First day of week:
 Start time of week:
 First week of year:
☐ Use 4 numbers for the year

0 1 2 1 OK Apply Cancel

Naming and administration of data files must be configured prior to the first recording. In order to configure these settings, click on the "Files" sub-branch of the corresponding data storage in the data storage dialog. On this level, you can configure file name and storage location.

6.7.1 Data file name

Base file name

Enter a file name prefix which is always the same, e.g. a term with relation to the data storage creating the files. The base file name will be supplemented automatically by an incrementing number if more than one data file with the same name exists.

Check the “Add base file name” selection box if you want the base file name to be also (part of) the file name. Uncheck it if you want other information to be shown in the file name.

However, it is always advisable to enter a base file name even if you do not intend to use it. In case, for instance, you use the text signal only for naming the files and the text signal is not available due to an error, *ibaPDA* will use the base file name instead (plus numbering, if necessary).

Optionally, you can also select other information to be part of the file name:

- File number
- Start date and time
- Start trigger name
- Text signal

Simply enable the option you wish to be added to the filename.

Below the base file name, an example of a file name is shown in blue letters. Whenever you enable or disable the naming options, you can immediately see the effect it would have on the next file name.

You cannot change the order of the optional information in the file name. If you want to customize the data file names differently, you should use the text signal.

Add file number

If this option is enabled, the file number will be added based on a sequential number between 0 and the maximum number of files.

- Maximum number of files
This setting specifies the number of the files provided for the recording. If the maximum number of files is reached, the next recording will overwrite the first file with the same name (then, e.g., again with index 0) unless other restrictions have been specified in the settings under "Cleanup strategy".
- Next file number
The index of the next file created. If necessary, this index can also be set manually on any value between 0 and the maximum number of files.

Add start date and time

If enabled, the current date and time will be added to the file name.

The format is `_yyyy-mm-dd_hh.mm.ss`

Please note that this is the date and time at the start of the data file. The values may differ from the Windows file date, which is the date at file closure.

Add start trigger name

This option is only available if the start trigger pool is used as trigger mode.

If enabled, the name of the trigger signal from the start trigger pool which fires first will be added to the file name. So, you can easily see which trigger event started the recording.

Add text signal

If you choose this option, a text signal section selected from the dropdown list nearby will be added to the file name.

In case you disable all other naming options, the data file will be stored with a name derived from a text signal and the file will be saved under this name.

If you check this option, additional options and an entry field are activated:

- In the field right next to the selection box, you can select the appropriate text signal from a dropdown list. Of course, the text signal must have been defined in the I/O Manager beforehand.
- Select whether the contents of the text signal should be read at file opening (= start trigger) or at file closing (= stop trigger). Which selection is best depends on the process you are measuring, i.e. the time the correct value of the text signal is available.
- Reuse text signal value
If this option is disabled, the text signal value will only be used once right after reception. If no new text signal value is received for the next data file, the base file name will be used instead.
By enabling this option, you determine that the last received text will be used again for the next data file. In this case, *ibaPDA* automatically supplements the file name by an index to prevent overwriting of the last file.
- Remove all blanks in the file name
When a text signal is configured with the “Fixed width” separating mode, it may happen that some characters are filled in with spaces. These blanks might have a negative effect on the a file name. Therefore, you can enable this option in order to remove blank characters in the file name.

6.7.2 Data file location

Base directory

The base directory is the main directory where the data files are stored. There is a default path for timebased data stores:

`c:\iba\Datastores`

The system will create automatically a sub-folder with index for each configured data store (Store1, Store2 etc.)

You may change the base directory. Then, enter the main path for storing the data files of this data storage. A valid drive and path must be entered. You may use the browser button <...> to find available drives and paths.

You can either select a local or network drive.

Note



For example, if you enter drive C:\, the local hard disk of the server computer is addressed. The browser shows the perspective of the server in terms of drives and paths.

Note



Basically, you can write the data files directly on a network share or NAS. However, the performance will be significantly lower, compared to the local harddisk.

Thus, iba recommends always configuring the local harddisk as base directory. By means of the software *ibaDatCoordinator* the terminated data files can be copied or moved automatically and reliably to another storage medium.

If, even so, you want to store the data files on a remote drive of another computer, you need to configure the user account which is used for running the *ibaPDA* server on the server computer also on the remote computer. This user account should have administrative rights. If it is not possible to set up the same account on both computers, you may enter *user name* and *password* of another user with the appropriate rights on the remote computer. *ibaPDA* then uses this login information to log on the remote computer in the attempt to write data files.

User name and password

You only have to enter this login information if you want to store the data files on a remote computer, on which the user account used by *ibaPDA* on the server computer is not configured (see above).

<Check path> Button

This button may be used for checking the base directory as entered above in terms of availability under consideration of the user account data.

Backup directory

In this field, you may enter another drive or path for file storage, just in case the base directory is not available, e. g. due to network problems. If *ibaPDA* cannot write on the base directory, it

automatically tries to write on the backup directory. So, it is advisable to enter a path to a local drive on the server computer as backup directory. The system checks every 15 minutes if the base directory is available again. If so, *ibaPDA* will store the next file in the base directory.

6.7.3 Subdirectory organization

Subdirectories can be used for a more structured storage of the data files. For every option and combination you choose, an example of the path is given immediately in blue letters.

None

The files are stored in the path specified in *Base directory*.

Hour, day, week or month based

Further subdirectories are added to the path specified as base directory. These contain parts of a complete file series. The names of the subdirectories are automatically derived from the date and/or time, depending on your selection. Each subdirectory then contains those files that were recorded in the relevant period of time.

The format of the subdirectories follows the pattern

- `yymmddhh` or `yyyymmddhh` for an hourly, daily or month-based setup
- `yyww` for a week-based setup

A "daily" subfolder, for example, would be named as follows: 050221 for February 21, 2005. The next "Daily" folder would then be 050222 and so on.

In case of a "Weekly" storage type, the folders would be named according to the calendar week number. For determining the week number, the Windows calendar is used. Since the week number may vary depending on region and country, several parameters need to be set:

- First day of week (e. g. Germany = Monday, USA = Sunday)
- Start time of week (the time at which the first day of the new week begins)
- First week of year (the week around January 1, the first complete week of the year or the week with four days in the new year).

You should adjust these settings to your regional conditions in order to get a proper week counting.

Maximum subdirectories

This setting is only relevant if a subdirectory organization was selected. It determines the maximum number of subdirectories according to the number of hours, days or months in the base directory.

Example:

"Hourly" organization, number of subdirectories = 2; up to two folders are created (these contain the file series); when the second hour has passed, a third folder is initially created; however, the first folder with all its contents will be deleted automatically, so that only two folders exist.

Use multiple levels of subdirectories

Check this option if you prefer separate subdirectories for year, month, week, day and hour.

The format of the subdirectories then follows the pattern

- `(yy) yy\mm\dd\hh` for an hour-, day- or month-based organization
- `(yy) yy\ww` for a week-based organization

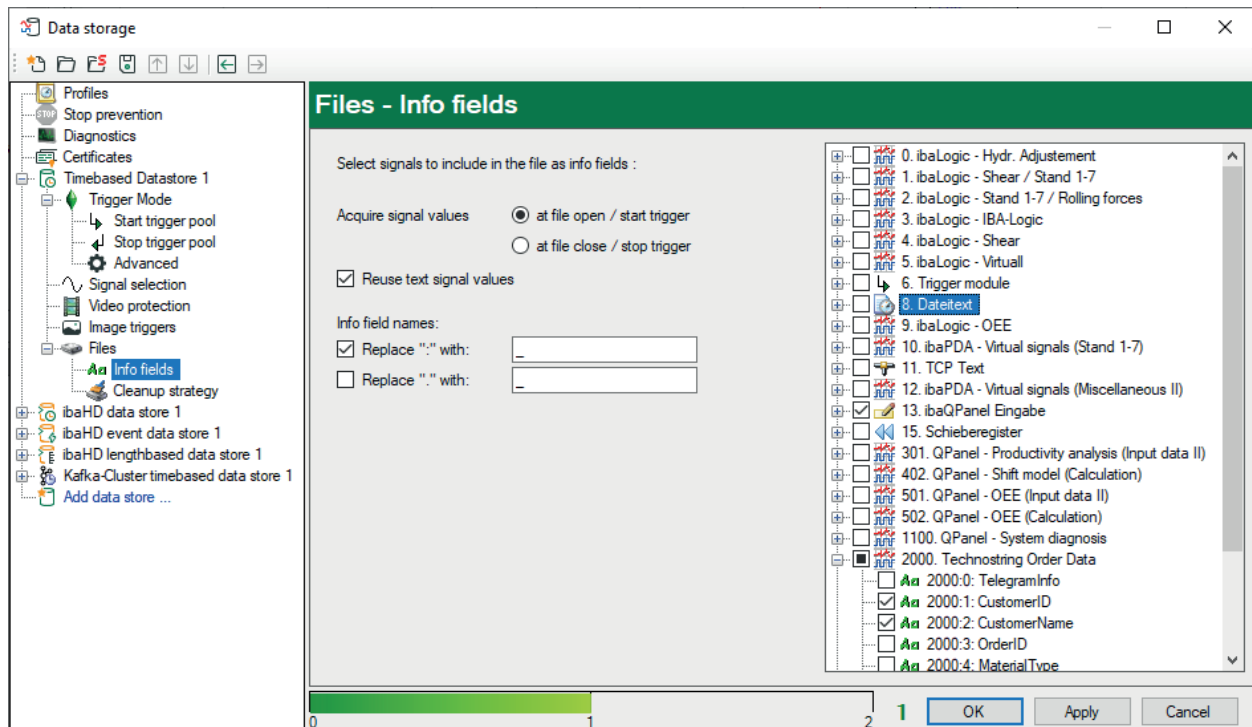
Restart file number when subdirectory changes

If this option is enabled, the file number will start again at 0 in every new subdirectory.

Example:

If the length of a data file is set to 1 hour with the storage type set to "Daily", you can use the file numbering for naming the file. Enable the option for resetting the file number in order to browse every "daily" subdirectory for file names ranging from xyz_0 to xyz_23.

6.7.4 Files - Info fields



Click on the *Info fields* node of the *Files* branch of the corresponding data storage to open the dialog. Here, you can set the signals (analog, digital or text signals) to be stored as info fields in the data files and the time they should be acquired.

Select signals to include in the file as info fields:

In the right part of the signal tree, you will find all modules with analog, digital and text signals as you defined them in the I/O Manager.

Enable the selection boxes in order to select individual signals or complete modules to be stored in the data file.

When analyzing with *ibaAnalyzer*, you will find the information of the info fields in the info header of the data file.

In the info fields, only one current value per data file is entered and saved, either at the start or at the end of the data file recording (see below).

Acquire signal values

By choosing this option, you determine when the values are used by an analog, digital or text signal for recording.

- At file open / start trigger: The current signal values are transferred into the data file at the start of a recording. Signal value changes during a recording process are taken into consideration for the next file at the earliest.
- At file close / stop trigger: The signal values are first transferred into the data file at the end of the recording. Signal values can be received, changed and sent again during a running recording. The last signal value received before closing the file will be taken. This option should be selected if the desired information is not available at the start of the recording.

Reuse text signal values

This option prompts *ibaPDA* to keep the received text information after reading instead of deleting it as usual. So, you make sure that new data files are supplied with the same text signal values in case no new text has been received.

When using the text signal for naming the file, *ibaPDA* automatically adds an incrementing index in order to provide unambiguous data file names.

Info field names: Replace ":" with ... and replace "." with ...

The use of ":" and "." in info fields can cause problems in the course of further processing and analysis of the data by other systems. In *ibaPDA*, these characters are used, e.g., in the signal IDs (analog [Module:Channel], digital [Module.Channel]).

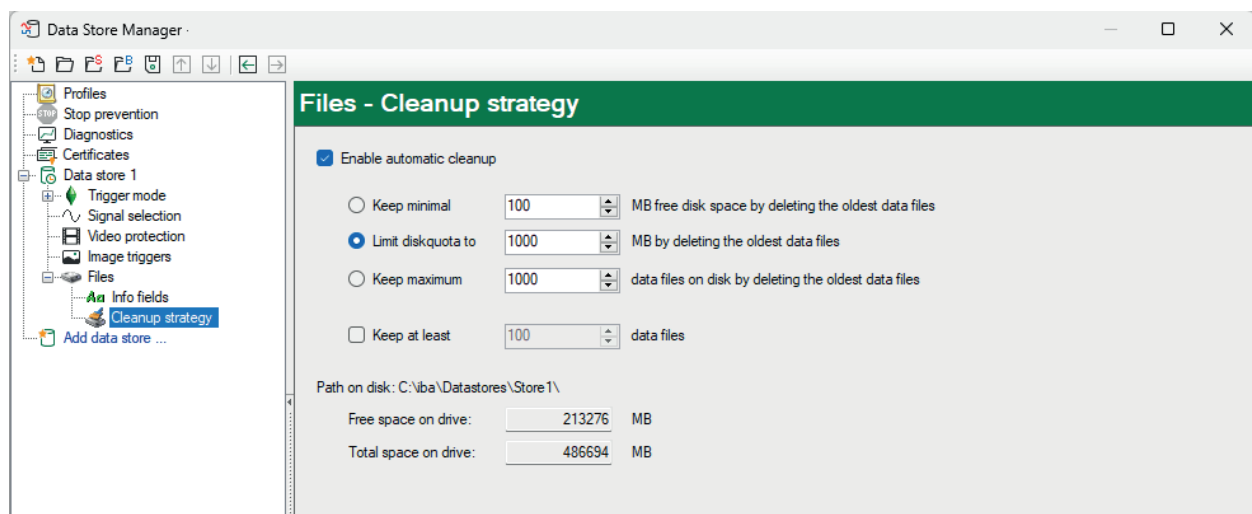
To avoid such problems, you can have the characters ":" and/or "." replaced by other, permissible characters here.

Enable the desired option and enter the new character in the field next to it.

6.7.5 Data files - Cleanup strategy

ibaPDA offers the possibility to check the available space on the storage medium and to initiate cleanup measures, if required.

Click on the "Cleanup strategy" node of the "Files" branch of the corresponding data storage to open the dialog.



We strongly recommend enabling this option in order to prevent the hard disk (HD) from "spilling over" together with the risk of system interruptions unless you use other methods to ensure always enough free disk space for new data files.

The cleanup strategy can be used to limit the memory for measured data.

The actual values of total and free memory space on the base directory drive is displayed in the center of the dialog window.

Enable automatic cleanup

Check this option in order to enable the automatic cleanup.

Keep minimal ... MB free disk space by deleting the oldest data files

If this option is enabled, the oldest files will be deleted in order to restore sufficient memory space as soon as the defined limit of free memory space is breached. Enter an appropriate value to ensure the system integrity.

The indication of the remaining free memory space refers to the total memory capacity of the drive, just as displayed under "Free space...".

Note



When using several *ibaPDA* systems that write on a common drive (file server), the permissible memory space should be the same for all *ibaPDA* systems. In this case, the "Limit disk quota..." option would be more preferable.

The same applies when running multiple data storages at the same time. Of course, you must enable this option for every data storage.

Limit disk quota to... MB by deleting the oldest data files.

This option limits the total disk space used by all dat files of the corresponding data storage in the base directory, including subdirectories. Select this option and enter a sufficient value for the disk space the expected data files presumably require.

Basically, we recommend using this option.

Keep maximum ... data files on hard disk by deleting the oldest data files.

If this option is enabled, the system will save as many data files as determined in this setting. In case this limit is exceeded and there is no other rule causing a deletion, the oldest data files will be overwritten. The number of files may fall below the specified number, but it can never exceed it.

Keep at least ... data files

If this option is enabled, as many old files as set here are saved from being deleted. The number of files refers only to the files created by the respective *ibaPDA* system.

This is particularly helpful, for example, if two *ibaPDA* systems or several recordings write on the same hard disk at the same time. In the long term, the number of files prevents that one of the two *ibaPDA* systems or data storages "goes short" in case only the MB limit is enabled.

This option can be activated in addition to the first two options, however, it has a higher priority.

6.7.6 Post-processing

As of *ibaPDA* v7.3.0, the *Files – Post-processing* branch is no longer available for configuring new data stores.

Instead, we recommend using the program *ibaDatCoordinator* for the automated further processing of data files. This software offers a diverse suite of features and also ensures greater reliability.

In cases where the post-processing feature was configured using a software version <v7.3.0, which was then upgraded to version ≥v7.3.0, the feature will still be available.

However, you must now also enter the user name and password in the configuration. The post-processing command is no longer executed automatically on the basis of the user profile associated with the *ibaPDA* service.

Instead, a dedicated user must be created who is authorized to perform post-processing.

6.8 Apply new data storage configuration

You can set up multiple data storages and profiles without leaving the dialog in between.

After you have finished the setup, click on the <Apply> or <OK> button.

Note



Whenever you have changed the configuration of one or more data stores you should close the configuration dialog with the <OK> button. At this time, the recording of data stores which were subject to change will be stopped and closed. Other, unchanged data stores will continue to record. The recording of the changed data stores will restart automatically with the new configuration. In any case, the samples will be retained. In accordance with the storage profile, the new data stores will directly start with the next sample.

Important: Changes in the data storage configuration will not affect the data acquisition of the server. The server continues the data acquisition even if the recordings are stopped and restarted.

6.9 Configuration file of data storage

All configuration data is stored in an XML file. The *ibaPDA* system uses three different configuration files for I/Os, data storage and view (layout). The *CurrentDataStoreConfig.ds* file is used for data storage configuration. The file is stored in a sub-folder of the *ProgramData* folder (path: `c:\ProgramData\iba\ibaPDA...`).

This file is loaded with each start of the system. It is updated whenever you change the configuration or leave the configuration dialog of the data storage by click on the <OK> button. The current configuration is therefore always stored in this file without explicitly saving it. Each time a new configuration is applied, the previous one is automatically saved in the `...\Backup` folder. The last ten configurations are always available in the backup folder.

Tip



You can access the saved configuration files in the backup folder directly by a special button in the toolbar of the data storage manager:



If the most recently used configuration file is not found, the server automatically loads the most recent backup configuration.

If you have applied a valid and functional configuration, it is recommended to also store the configuration file under another name in a protected folder or disk of your choice in order to avoid unintended modifications or loss of configuration data. You can also use the *Save project...* function in the *File – Save project...* menu.

Saving a data storage configuration

1. Open the data storage dialog after making sure that the current configuration is correct.
2. Click the button  (save configuration) in the toolbar of the data storage dialog.
3. In the dialog box, select the path where the file should be saved. We recommend choosing the path `c:\ProgramData\iba\ibaPDA`, as this is the default path of the data storage configuration files. But of course you may choose any other folder.
4. Enter a file name (prefix).
5. Enter a file extension (.ds or .zip).
If you select .ds then a single `file name.ds` file is created. If a file already exists with the same name then it is overwritten.
If you choose .zip then a `filename.zip` ZIP file is created containing a `CurrentData-StoreConfig.ds` file.
If a file already exists with the same name, the file within the ZIP file will be replaced. This also applies if the ZIP file contains an *ibaPDA* project file in which more files are stored.
6. Click on <Save>.

Loading an existing data storage configuration

1. Open the data storage dialog.
2. Click the button  (open) in the toolbar of the data storage dialog.
3. Select the desired configuration file by using the browser. Usually, you will find the configuration files in the folder `c:\ProgramData\iba\ibaPDA\Server`. The browser only shows files with the `.ds` extension.
The `CurrentDataStoreConfig.ds` file contains the latest configuration.
4. Press <Open>. The configuration will be loaded and compared with the signals, drives and the license information available in the dongle. If no conflicts occur, the configuration is considered to be valid and the recording starts.

Loading an existing configuration file in simulation mode

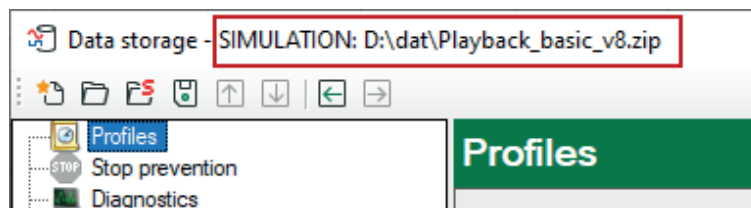
Similar to the I/O Manager, it is also possible to load a data storage configuration in simulation mode during data recording. A data store can therefore be configured without the I/O configuration being accepted by the server. The iba support department can better diagnose user problems by using the simulation mode because the configuration is displayed exactly as on the user's original system.

To load a data storage configuration into simulation mode, use the appropriate button in the toolbar of the data storage configuration dialog:



In the following dialog, you can choose the desired configuration file (.ds) or load an *ibaPDA* project file (.zip).

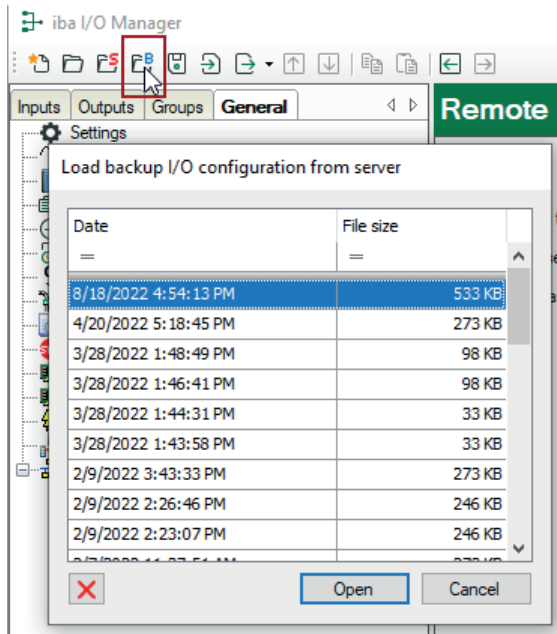
If the file is loaded, "SIMULATION" and the full path of the file are displayed in the header of the dialog. The <Apply> and <OK> buttons are not available.



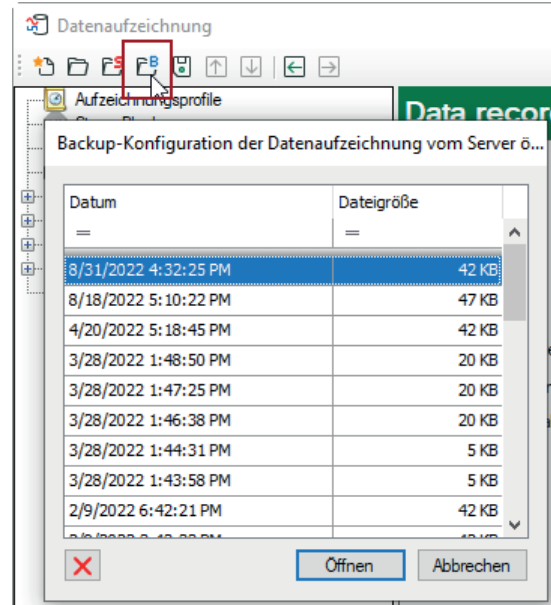
Alternatively, you can drag the file from Windows Explorer into the tree of data storage dialog in order to activate simulation mode.

6.10 Loading a backup configuration file

Each time a new I/O configuration or datastore configuration is applied, *ibaPDA* automatically creates a backup of the previous configuration. These backups are stored in the path `%ProgramData%\iba\ibaPDA\Backup` on the *ibaPDA* server computer. In the toolbar of the I/O Manager or data storage manager there is a dedicated button which provides direct access to the backups.



Open backup in I/O Manager



Open backup in datastore manager

If you click on this button, a dialog window will open, which shows a list of the available backups with their date and time of creation and file size.

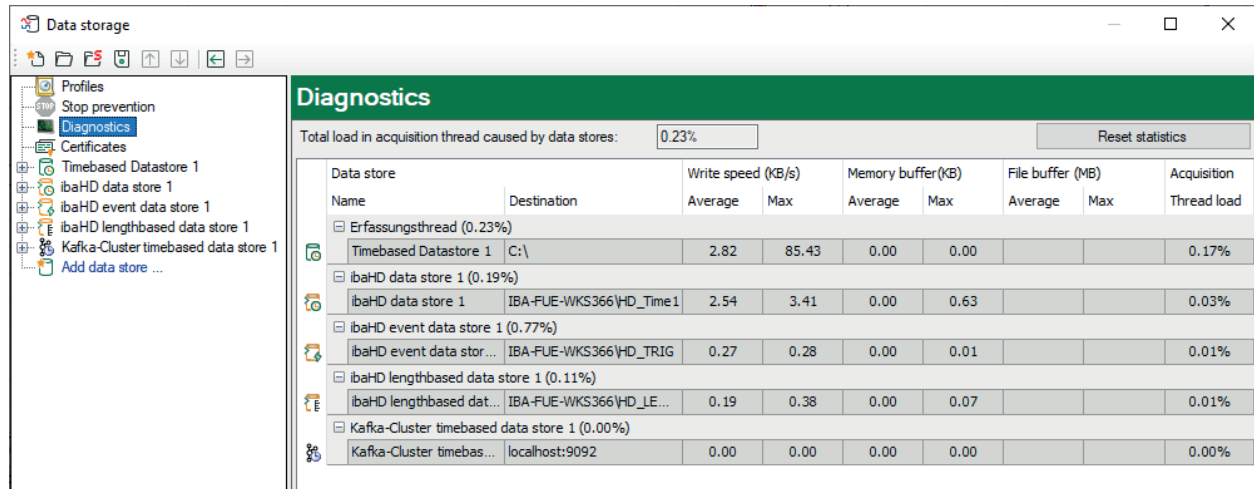
By selecting a row and clicking on <Open> or by doubleclicking on a row you can open the corresponding backup in the I/O Manager or data storage manager respectively.

If you want to apply the configuration from the backup, then click on <Apply> or <OK>.

In the dialog window with the backup list there is a button to remove a selected backup, if needed.

7 Diagnostics for data stores

In the tree of data stores, there is one *Diagnostics* node, which provides some information about the system load due to the data stores. The measurement must be running meanwhile.



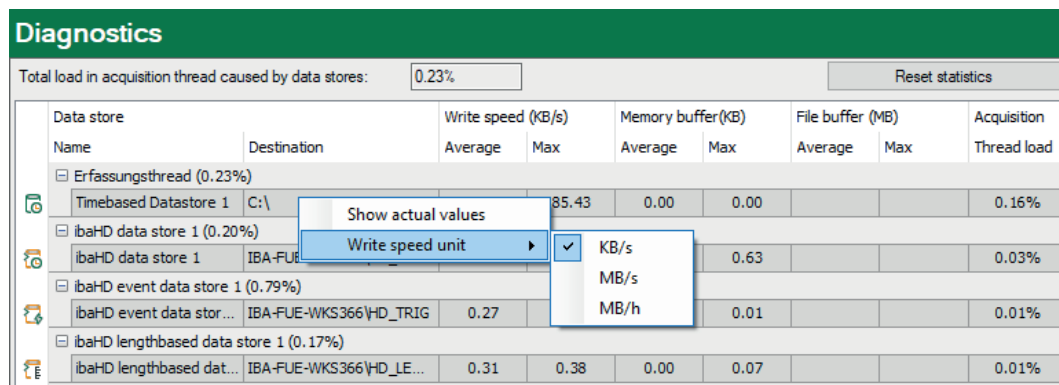
The screenshot shows the 'Diagnostics' window with a table of performance metrics. The table is grouped by threads. The first group is 'Erfassungsthead (0.23%)' with one row: 'Timebased Datastore 1' with a thread load of 0.17%. The second group is 'ibaHD data store 1 (0.19%)' with one row: 'ibaHD data store 1' with a thread load of 0.03%. The third group is 'ibaHD event data store 1 (0.77%)' with one row: 'ibaHD event data stor...' with a thread load of 0.01%. The fourth group is 'ibaHD lengthbased data store 1 (0.11%)' with one row: 'ibaHD lengthbased dat...' with a thread load of 0.01%. The fifth group is 'Kafka-Cluster timebased data store 1 (0.00%)' with one row: 'Kafka-Cluster timebas...' with a thread load of 0.00%.

Data store		Write speed (KB/s)		Memory buffer (KB)		File buffer (MB)		Acquisition
Name	Destination	Average	Max	Average	Max	Average	Max	Thread load
Erfassungsthead (0.23%)								
Timebased Datastore 1	C:\	2.82	85.43	0.00	0.00			0.17%
ibaHD data store 1 (0.19%)								
ibaHD data store 1	IBA-FUE-WKS366\HD_Time1	2.54	3.41	0.00	0.63			0.03%
ibaHD event data store 1 (0.77%)								
ibaHD event data stor...	IBA-FUE-WKS366\HD_TRIG	0.27	0.28	0.00	0.01			0.01%
ibaHD lengthbased data store 1 (0.11%)								
ibaHD lengthbased dat...	IBA-FUE-WKS366\HD_LE...	0.19	0.38	0.00	0.07			0.01%
Kafka-Cluster timebased data store 1 (0.00%)								
Kafka-Cluster timebas...	localhost:9092	0.00	0.00	0.00	0.00			0.00%

The performance values of all data stores are shown in the table. There is one row per data store. The rows are grouped according to the threads that write the data.

In each group row is the name of the thread and (in brackets) its share of the load. The load average is displayed by default. But, you can switch between the average and actual value using the context menu.

In the same menu, you can also adjust the display unit of the write speed.



The screenshot shows the 'Diagnostics' window with a context menu open over the 'Timebased Datastore 1' row. The menu has two options: 'Show actual values' and 'Write speed unit'. The 'Write speed unit' option is selected, and a sub-menu is visible showing three options: 'KB/s' (selected), 'MB/s', and 'MB/h'.

Data store		Write speed (KB/s)		Memory buffer (KB)		File buffer (MB)		Acquisition
Name	Destination	Average	Max	Average	Max	Average	Max	Thread load
Erfassungsthead (0.23%)								
Timebased Datastore 1	C:\	2.82	85.43	0.00	0.00			0.16%
ibaHD data store 1 (0.20%)								
ibaHD data store 1	IBA-FUE-WKS366\HD_Time1	2.54	3.41	0.00	0.63			0.03%
ibaHD event data store 1 (0.79%)								
ibaHD event data stor...	IBA-FUE-WKS366\HD_TRIG	0.27	0.28	0.00	0.01			0.01%
ibaHD lengthbased data store 1 (0.17%)								
ibaHD lengthbased dat...	IBA-FUE-WKS366\HD_LE...	0.31	0.38	0.00	0.07			0.01%

In addition to the write speed, you will also see information on the occupancy of the main memory ("memory buffer") and the file memory ("file buffer"). Values in the "File buffer" columns only exist for HD data stores and for databases, cloud and MQTT data stores if the file buffer has been enabled for these data store types. For further information, please refer to the manuals for ibaHD-Server and the data stores (Data-Stores).

Depending on the data store type, the data displayed can be interpreted differently:

Normal time-based data stores with synchronous file access

The rows of this store type show the write speed of the data on the hard disk. The columns for the buffered data show zeros because the data are written directly on the disk. The column of the acquisition thread load shows the proportion of time needed for the run-length encoding and writing on the disk.

Time-based data stores with asynchronous file access

The rows of this store type show the write speed of the data on the hard disk. The columns for the buffered data show the amount of data that still has to be written on the disk. The column of the acquisition thread load shows the proportion of time needed for the run-length encoding. The actual writing process is carried out in the write thread, the load of which you can see in the group row.

ibaQDR data storage

The rows of this type of store show the write speed of all the input coil files from all measuring locations. The write speed of product coils and input product coils is not displayed. The group rows of the QDR product thread show the load due to the generation of the product coil files. The columns for the buffered data show the amount of length-based data that has not yet been used for a product coil file. The column of the acquisition thread load shows the share of the load due to saving time-based data in input coil files and generating length-based data at all measuring locations.

ibaHD data stores (time, length and event-based)

The rows of this store type show the write speed of the data on the ibaHD server. The columns for the buffered data show the amount of data that still has to be written onto the ibaHD server. These values should usually be close to zero. If the connection to the ibaHD server is interrupted, these values increase rapidly. If the file buffer has been activated for HD recording and the normal buffer is full, the data is stored in files on the hard disk ("file buffer").

The column of the acquisition thread load shows the proportion of time needed to provide the data to be written on the ibaHD server. This period includes:

- Run-length encoding for time-based stores
- Event trigger calculation for event-based stores
- Generating length-based data for length-based stores

DB/Cloud, MQTT and MindSphere data stores

For this storage type, the rows show the write speed of the data to a DB or cloud server or an MQTT broker. The columns for the buffered data show the amount of data that still needs to be written to the disk. These values should usually be close to zero. If the connection to the target server or broker is interrupted, these values increase rapidly. If the file buffer has been activated for HD recording and the normal buffer is full, the data is stored in files on the hard disk ("file buffer").

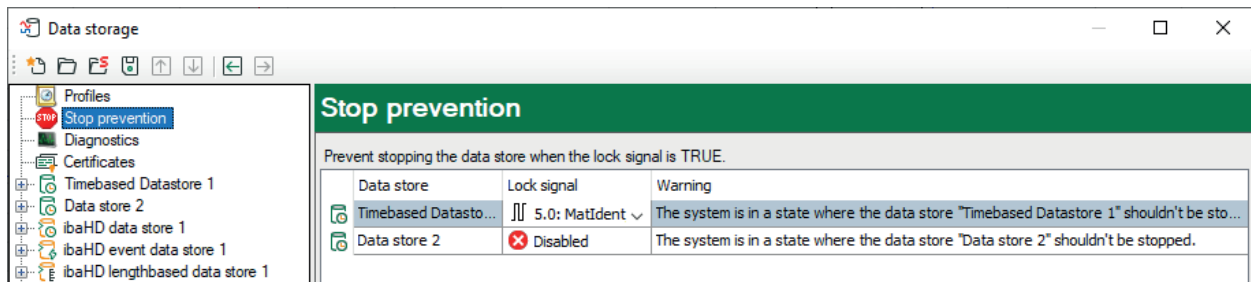
The column for the acquisition thread load shows the proportion of time needed for the evaluation of the triggers and generation of the row data.

At the top, above the table, the total load of the acquisition thread due to the data store is displayed. This is the sum of all the values in the *Acquisition thread load* column. It is a measure of how much time is needed to process the data which should be stored. If this value increases to 100%, alarm generation and client updates are suspended. If the value remains at 100% for a long time, the *ibaPDA* system becomes overloaded and acquisition is stopped when the driver buffer overflows.

Tip

With the *DataStoreInfo...* functions in the expression builder, you can make the relevant diagnostic data available as virtual signals for each data storage type. You can then visualize and record these signals.

8 Stop prevention



By enabling the “Stop prevention” function you have the possibility to prevent the data storage from being stopped by accident.

Stop prevention only affects only standard and QDR data stores (data files).

You can enable the stop prevention by selecting any digital signal. If attempts to stop the storage occur on an *ibaPDA* client while the signal is TRUE, a message box pops up asking for confirmation. The data storage will only be stopped if you answer this question with “Yes”. Using this feature you can protect a running data storage with signals from the plant, the process or by an operation.

If stop prevention is enabled, i.e., a lock signal has been assigned, a red STOP icon appears in the stop prevention node. If no lock signal is assigned, the icon is gray.

Lock signal

Click on the drop-down icon in this field and select the desired signal from the signal tree.

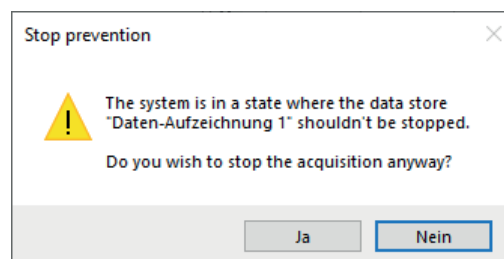
All digital signals can be used as lock signal, e. g.

- Digital input signals from sensors or PLC
- *ibaQPanel* inputs
- Virtual signals

Warning text

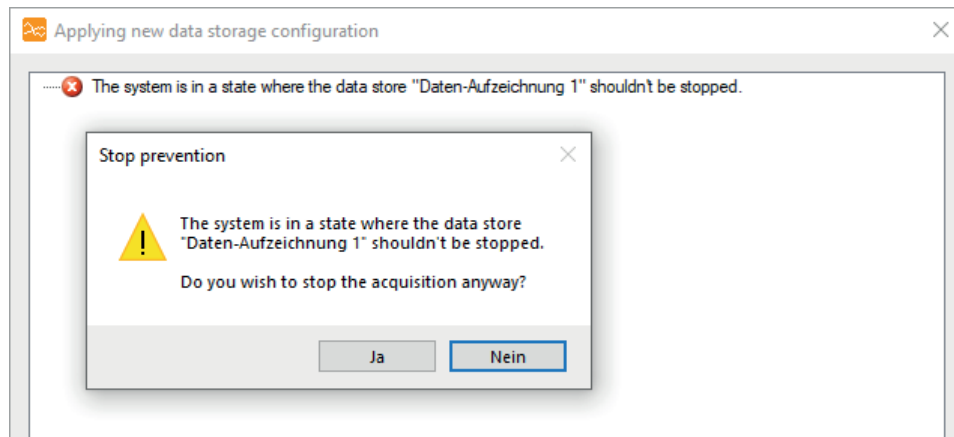
You can overwrite or modify the standard text in this field. This helps you to include practical advice, directions or contact data in the warning message.

If the stop prevention is triggered by clicking on the stop trigger button, the warning message pops up immediately.



You can only acknowledge this message. If you still want to stop the data storage, you must deactivate the stop prevention by deleting the lock signal. Alternatively, you can stop the acquisition, in which case the same message will also appear. Nevertheless, you then have the option to end the acquisition and thus also the storage.

If the stop prevention is triggered by a change of the data store configuration, the warning message pops up, supplemented by an entry in the validation dialog.



If you click <Yes> at this point, then the storage in question is stopped, whereas the new configuration is applied and the storage is restarted. Meanwhile, the acquisition continues.

9 Certificates

The *Certificates* node gives you access to the central certificate store. This node is available in both the I/O manager and the data storage configuration dialog. From both locations, you can access the same pool of certificates and can manage the relevant certificates.

A detailed description of the certificate store and the handling of certificates can be found in the *ibaPDA* manual part 2, chapter *Certificates*.

10 Support and contact

Support

Phone: +49 911 97282-14

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.

Product Security Incident Response Team (PSIRT) at iba

Current advisories: www.iba-ag.com/en/security

Email: psirt@iba-ag.com

Contact

Headquarters

iba AG
Gebhardtstrasse 10–20
90762 Fuerth
Germany

Phone: +49 911 97282-0

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