New Features in ibaAnalyzer 6.8.0

Author: Michael Verschaeve

Date: 6 July 2016

Table of contents

1 New functions

1.1 Acoustic vibration analysis functions

Some functions were implemented in order to facilitate analyses with the application domain of acoustic vibration analysis in mind. These functions were implemented in version 6.7.2 of ibaAnalyzer but haven't been documented in a new features document until now.

1.1.1 DbScale function

The *DbScale* function provides logarithmic scaling of a signal spectrum. The result is in Db. A reference value can be specified. The function results are only meaningful when applied to signals or expressions that represent an amplitude spectrum. The ibaAnalyzer functions that return an amplitude spectrum are:

- FftAmpl
- FftInTimeAmpl
- FftOrderAnalysisAmpl

Also calculations applied to these functions (e.g. *IntSpectrum* or *AWeighting,* see below) before applying the *DbScale* can produce meaningful results.

The function takes the following arguments:

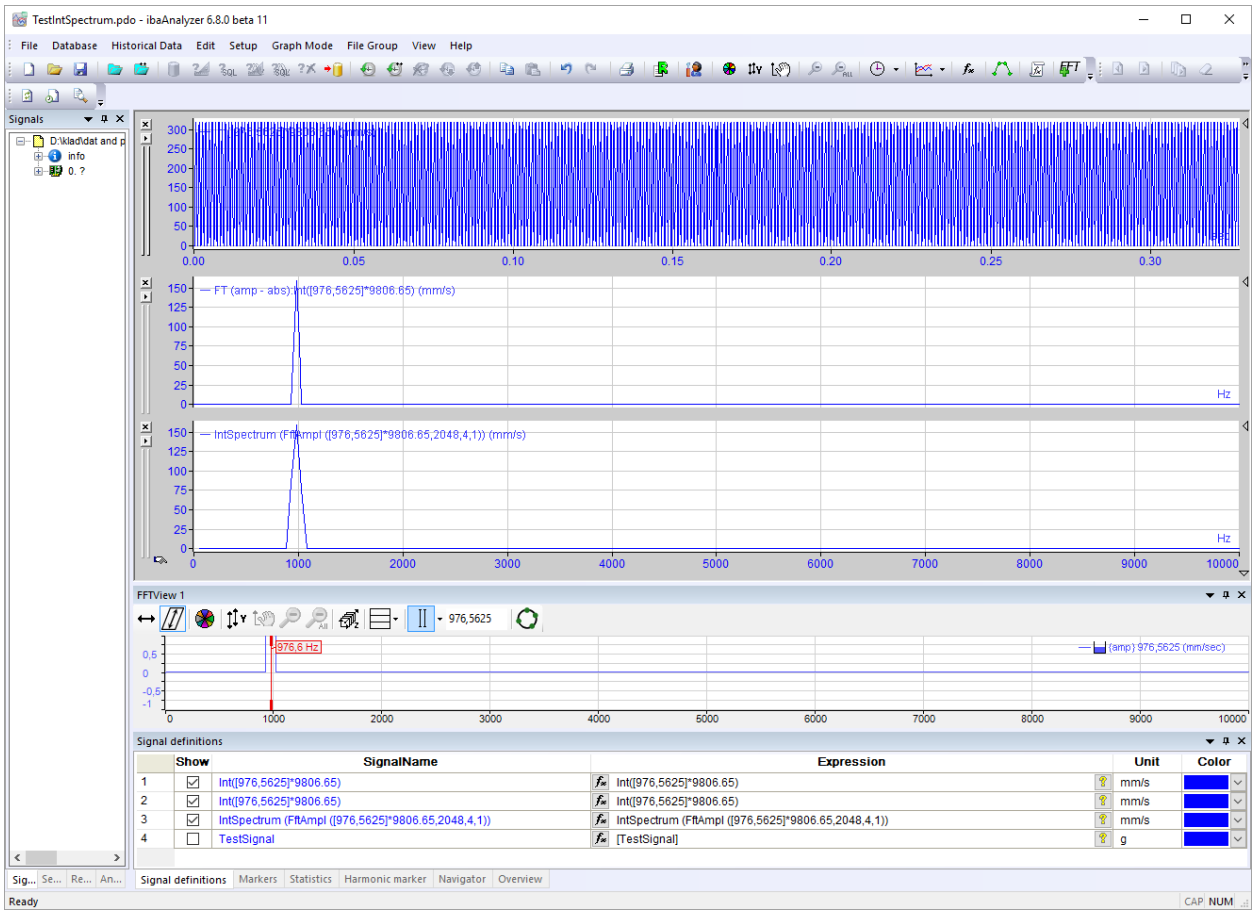
- Spectrum: The spectrum to apply Db scaling to.
- Reference: An optional reference value, the spectrum is divided by this value before the logarithm is taken. Hence this value should be the amplitude for which the Db scaling should return 0.

1.1.2 IntSpectrum function

The *IntSpectrum* function provides spectral integration of a signal spectrum. E.g. the frequency spectrum of an acceleration sensor (mm/s^2) can be converted to a spectrum of the vibration velocity (mm/s). The function results are only meaningful when applied to signals or expressions that represent an amplitude spectrum.

The function takes only the following argument:

- Spectrum: The spectrum to apply the spectral integration scaling to.

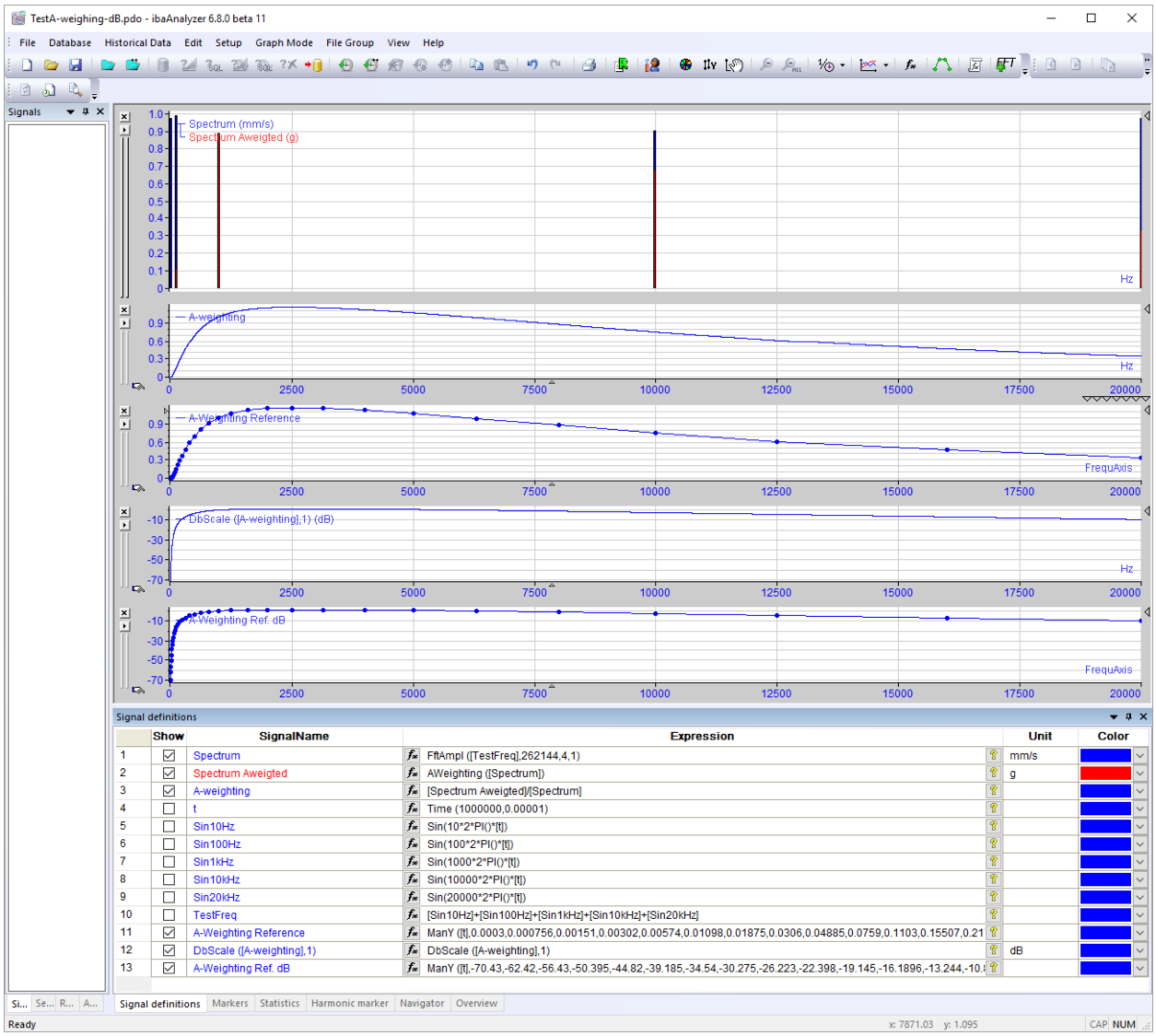


1.1.3 AWeighting function

The *AWeighting* function, applied to the spectrum of a sound signal is used for the detection of the evaluated sound pressure level. The result is a spectrum which is rated such as the human ear perceives the signal. The function results are only meaningful when applied to signals or expressions that represent an amplitude spectrum.

The function takes the following arguments:

- Spectrum: The spectrum to apply the A-weighting to.
- Type: An optional value to indicate what weighting function should be used. The parameter can take on the following values:
  - *0: use standard A-weighting. This is the default that is used when the parameter is omitted.*
  - *1: use B-weighting.*
  - *2: use C-weighting.*
  - *3: use D-weighting.*


TestA-weighing-dB.pdo - ibaAnalyzer 6.8.0 beta 11
File Database Historical Data Edit Setup Graph Mode File Group View Help
Signals
Spectrum (mm/s)
Spectrum Aweigted (g)
Hz
A-weighting
A-Weighting Reference
FrequAxis
DbScale ([A-weighting],1) (dB)
A-Weighting Ref. dB
Signal definitions
Show SignalName Expression Unit Color
1 Spectrum FftAmpl ([TestFreq],262144,4,1) mm/s
2 Spectrum Aweigted AWeighting ([Spectrum]) g
3 A-weighting [Spectrum Aweigted]/[Spectrum]
4 t Time (1000000,0.00001)
5 Sin10Hz Sin(10*2*PI()*[t])
6 Sin100Hz Sin(100*2*PI()*[t])
7 Sin1kHz Sin(1000*2*PI()*[t])
8 Sin10kHz Sin(10000*2*PI()*[t])
9 Sin20kHz Sin(20000*2*PI()*[t])
10 TestFreq [Sin10Hz]+[Sin100Hz]+[Sin1kHz]+[Sin10kHz]+[Sin20kHz]
11 A-Weighting Reference ManY ([t],0.0003,0.000756,0.00151,0.00302,0.00574,0.01098,0.01875,0.0306,0.04885,0.0759,0.1103,0.15507,0.21
12 DbScale ([A-weighting],1) DbScale ([A-weighting],1) dB
13 A-Weighting Ref. dB ManY ([t],-70.43,-62.42,-56.43,-50.395,-44.82,-39.185,-34.54,-30.275,-26.223,-22.398,-19.145,-16.1896,-13.244,-10.
Signal definitions Markers Statistics Harmonic marker Navigator Overview
Ready
x: 7871.03 y: 1.095
CAP NUM

1.2 Percentile functions

In previous versions of ibaAnalyzer, the *Percentile* function does not have the …*Moving* and …*InTime* variants while these variants are present for other statistical functions. This has been amended in the current ibaAnalyzer version.

1.2.1 PercentileInTime function

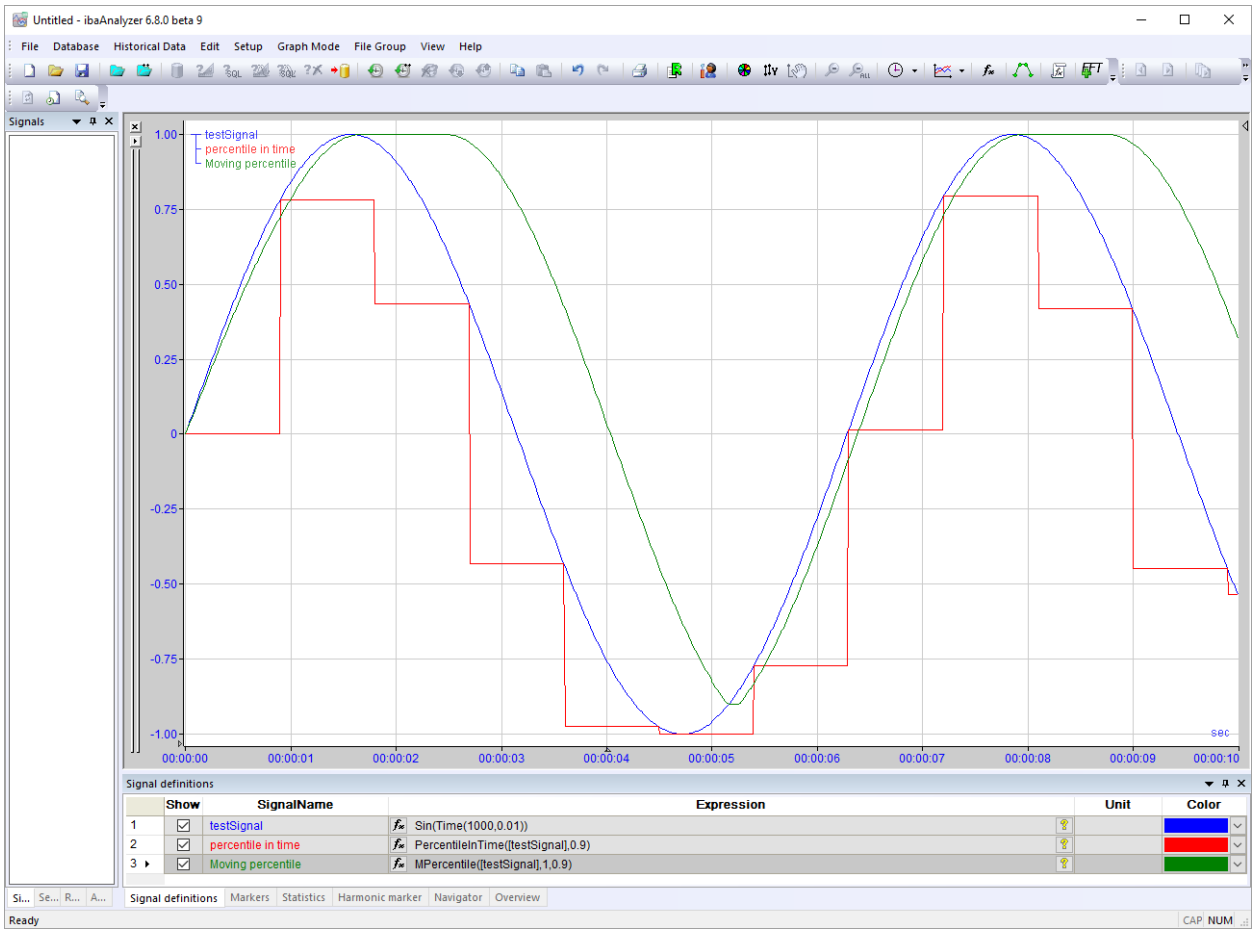
The *PercentileInTime* function splits the values of 'expression' along the X-axis in non-overlapping adjacent intervals and returns the requested percentile in each interval. The function takes the following parameters:

- Expression: The signal whose values generate the dataset to take the percentile from.
- Interval: The size of the intervals the percentile will be taken over, specified in X-axis units
- Percentile: Proportion of values that must lie below the reported result. Must be entered as a decimal value between 0 and 1. The parameter is optional, when omitted the value 0.5 is taken, which corresponds with the median value being reported.

1.2.2 MPercentile function

The *MPercentile* function is similar as the *PercentileInTime* function, but this time the intervals overlap for all but one measuring point. The interval the percentile is taken over can be envisioned to 'move' over the X-axis. The function takes the following parameters:

- Expression: The signal whose values generate the dataset to take the percentile from.
- Interval: The size of the intervals the percentile will be taken over, specified in X-axis units
- Percentile: Proportion of values that must lie below the reported result. Must be entered as a decimal value between 0 and 1. The parameter is optional, when omitted the value 0.5 is taken, which corresponds with the median value being reported.

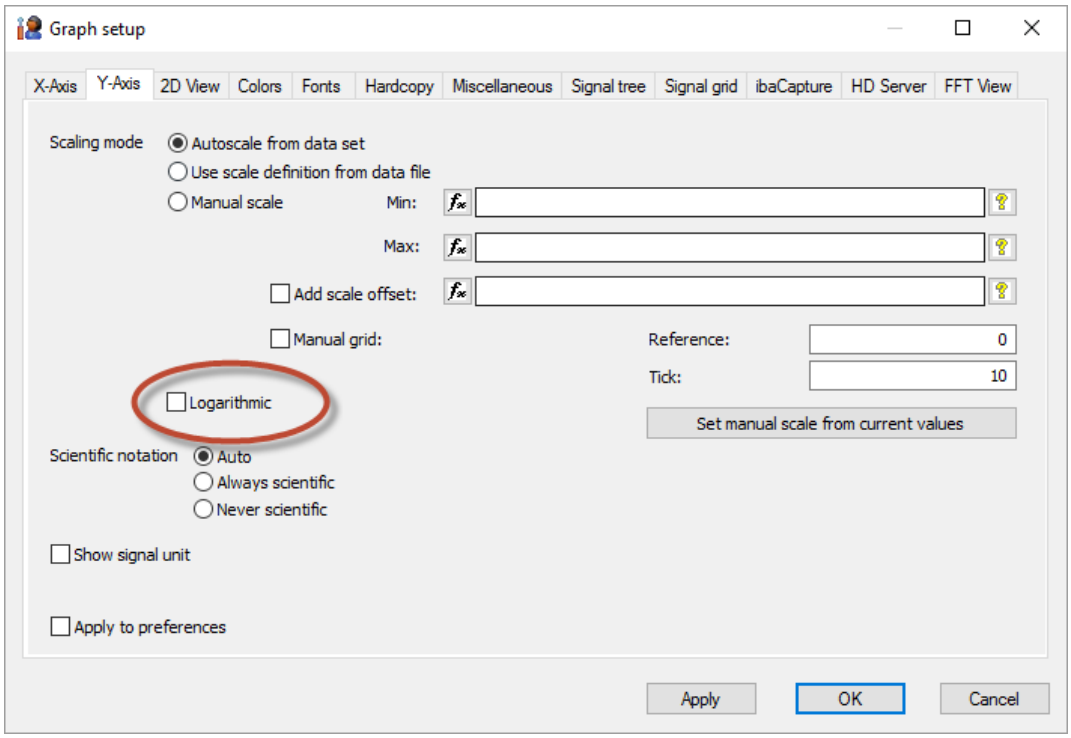

Untitled - ibaAnalyzer 6.8.0 beta 9
File Database Historical Data Edit Setup Graph Mode File Group View Help
Signals
testSignal
percentile in time
Moving percentile
1.00
0.75
0.50
0.25
0
-0.25
-0.50
-0.75
-1.00
00:00:00
00:00:01
00:00:02
00:00:03
00:00:04
00:00:05
00:00:06
00:00:07
00:00:08
00:00:09
00:00:10
sec
Signal definitions
Show SignalName Expression Unit Color
1 testSignal Sin(Time(1000,0.01))
2 percentile in time PercentileInTime([testSignal],0.9)
3 Moving percentile MPercentile([testSignal],1,0.9)
Signal definitions Markers Statistics Harmonic marker Navigator Overview
Ready
CAP NUM

2 Graphmanager

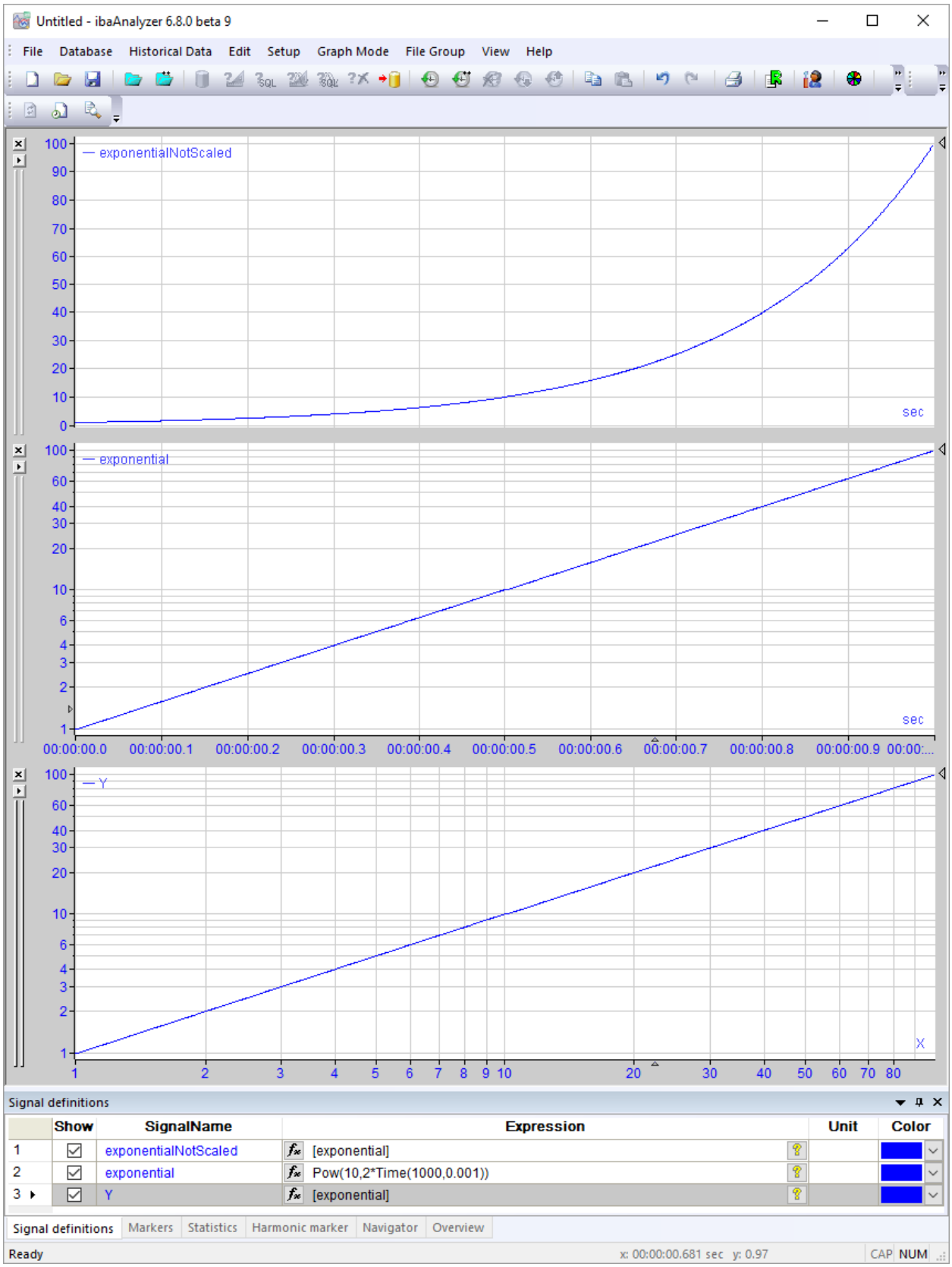
2.1 Logarithmic scaling of graphs

Standard graphs can be depicted on a logarithmic scale in the vertical direction, i.e. a logarithmic scale is applied to the Y-axis (value axis). For an XY plot, both horizontal and vertical logarithmic scaling can be applied.

The setting can be selected in the Y-axis settings in the Graph Setup:



For an XY plot, the setting is also available on the X-axis settings.


Untitled - ibaAnalyzer 6.8.0 beta 9
File Database Historical Data Edit Setup Graph Mode File Group View Help
exponentialNotScaled
sec
exponential
sec
00:00:00.0 00:00:00.1 00:00:00.2 00:00:00.3 00:00:00.4 00:00:00.5 00:00:00.6 00:00:00.7 00:00:00.8 00:00:00.9
Y
X
Signal definitions
Show SignalName Expression Unit Color
1 exponentialNotScaled [exponential]
2 exponential Pow(10,2*Time(1000,0.001))
3 Y [exponential]
Signal definitions Markers Statistics Harmonic marker Navigator Overview
Ready
x: 00:00:00.681 sec y: 0.97
CAP NUM

3 Database extraction

3.1 SQLite

All database functionality is now officially supported for SQLite except BLOB extract.

The connection to the SQLite database needs to happen through ODBC.

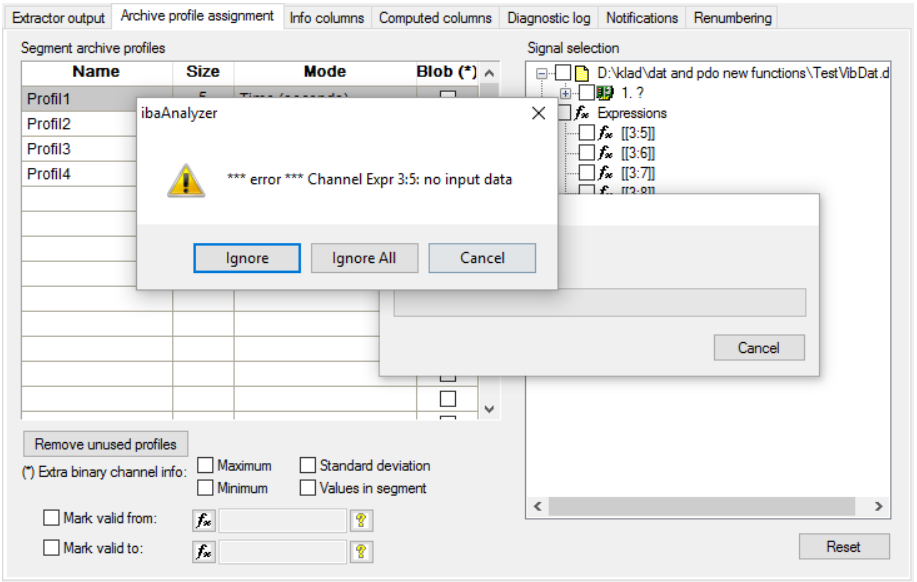
The ODBC drivers (32 and 64 bit) can be found here:

http://www.ch-werner.de/sqliteodbc/

When trying to extract into BLOB, an error will be given for each channel for which blob-extract was selected, the channel itself will not be extracted.

3.2 Error reporting

When errors happen during extracting interactively, one now has to option to dismiss all future errors during the extract or cancel the extract entirely. In previous versions of ibaAnalyzer, each error had to be acknowledged separately, which was an annoyance when for instance several hundred errors happened due to a wrong database setup.



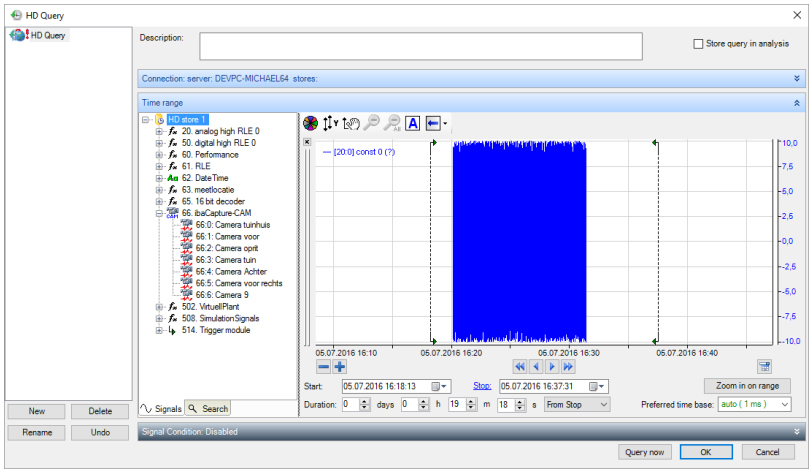
The error message box provides the following options:

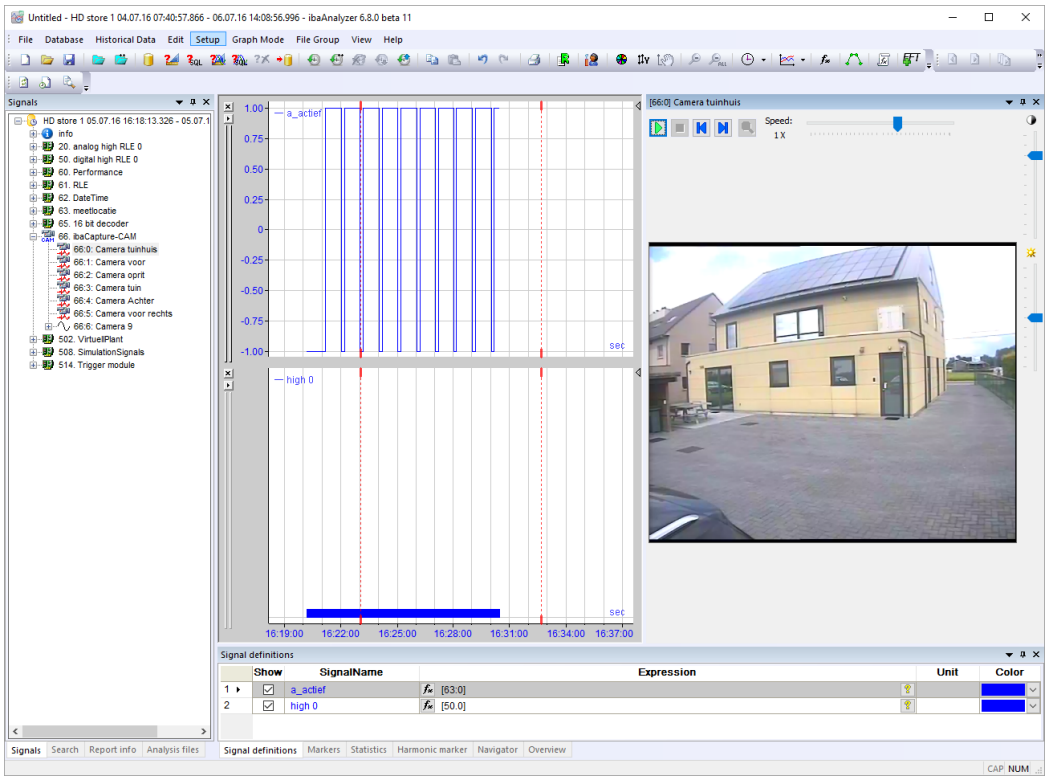
- Ignore: Dismiss only the current error, the extraction will continue and subsequent errors will be reported.
- Ignore all: Dismiss the current and all subsequent errors, the extraction will continue and no subsequent errors will be reported.
- Cancel: Dismiss the current error and abort the extraction.

4 ibaHD

4.1 ibaHD Capture CAM

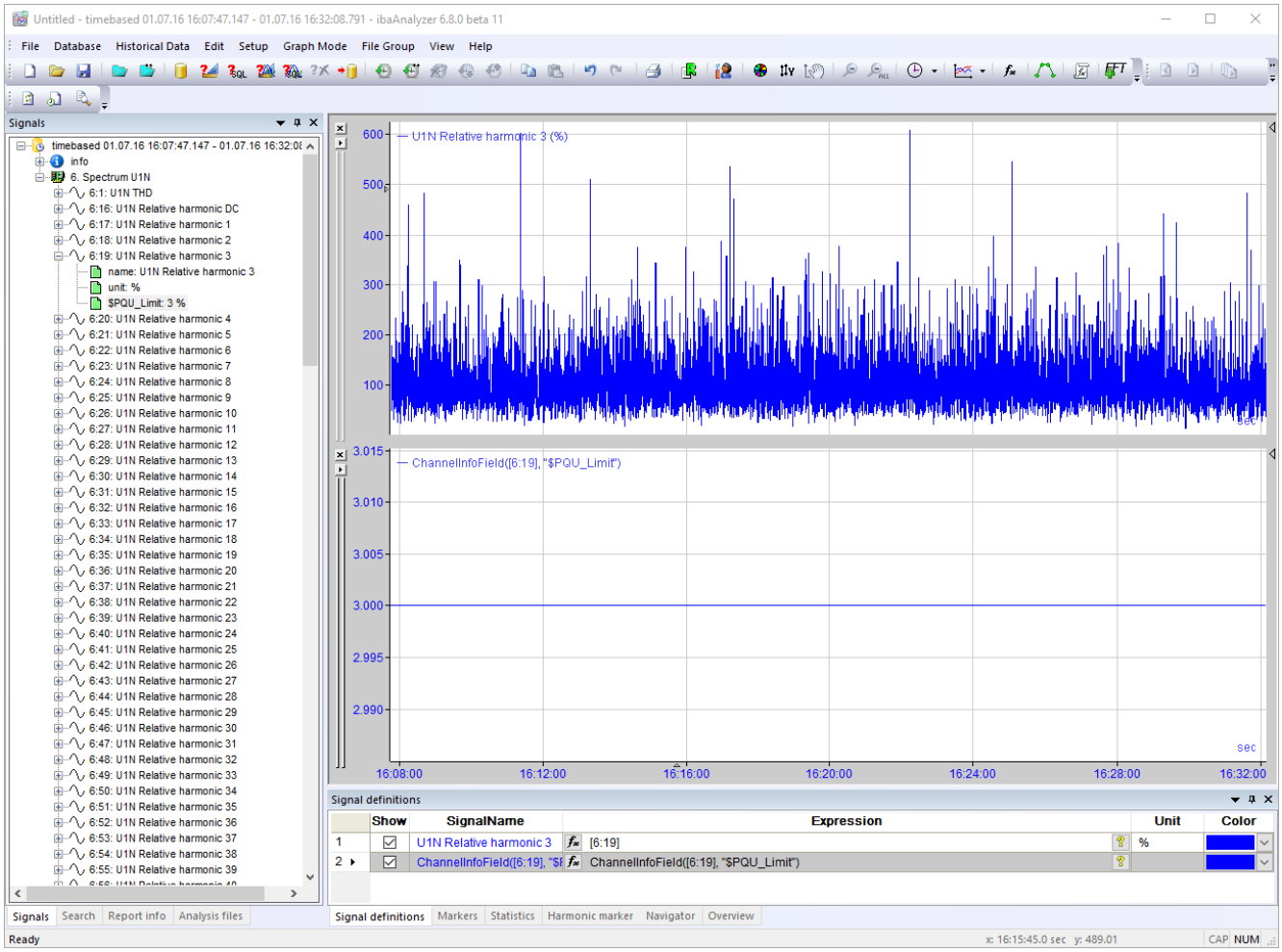
It is possible to store ibaCapture-CAM modules on the HD server since version 1.7.0 of the HD-server and version 6.36.0 of ibaPDA. In the current version of ibaAnalyzer, it is supported to query these modules. Hence ibaCapture-CAM videos can be shown and synced with the HD-data.





4.2 ibaPQU infofield

Since version 6.36.0 a channel infofield called "*$PQU_Limit*" is written to the HD server for all harmonic PQU channels. With the current version of ibaAnalyzer, this infofield will be present in the signal tree when available. Its value can also be used with the *ChannelInfoField* function.



5 64 bit support

A 64 bit version of ibaAnalyzer is available. The advantages of a 64 bit installation are:

- More memory can be allocated, allowing for larger analyses.
- Database extraction will work when only 64 bit versions of the OLE DB or ODBC drivers are available (which is the default case on a 64 bit system)

Two versions of the installer are available:

- ibaAnalyzerInstall_x86_v6.8.0.exe. This is the 32 bit version.
- ibaAnalyzerInstall_x64_v6.8.0.exe. This is the 64 bit version.

Theoretically, both versions can be installed in parallel on a 64 bit system, file associations and the links in the main menu will correspond with the last installed version, however, we do not recommend this practice.

All functionality of the 32-bit version of ibaAnalyzer is supported except:

- ibaCapture-HMI is not supported.
- ibaCapture-CAM is supported on condition that version 3.7.4 or later of the client software is installed.
- All database functionality is supported on condition that 64 bit versions of the required OLE DB or ODBC drivers are installed and configured properly.