New Features in ibaPDA v8.6.0

2024-01-24

iba AG • Königswarterstr. 44 • D-90762 Fürth • Germany • www.iba-ag.com

Table of contents

1 General remarks

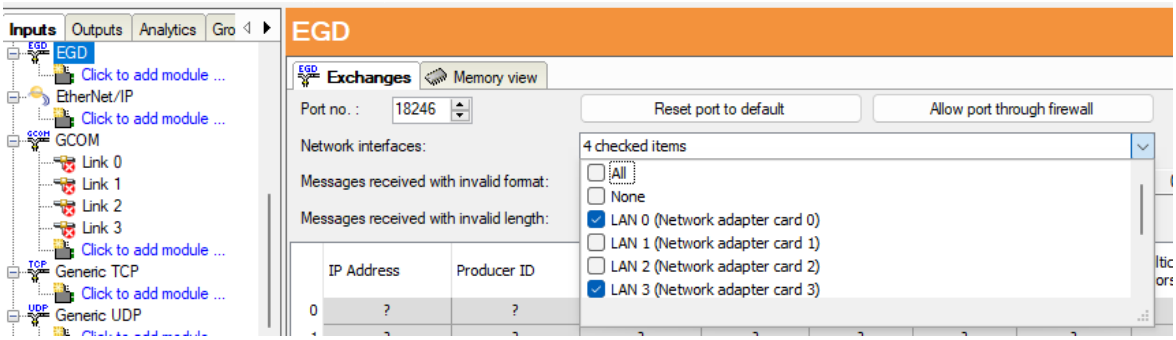
None

2 Changes to TCP/IP- and UDP-based interfaces

The following information relates to these interfaces:

- AN-X-DCSNet
- EGD
- EtherNet/IP
- Generic TCP
- Generic UDP
- ibaCapture
- ibaCapture-HMI
- ibaLogic TCP
- ibaNet-E
- Modbus TCP Server
- S7 TCP/UDP
- Sisteam TCP
- TDC TCP/UDP
- VIP TCP/UDP

In previous versions of ibaPDA, when opening TCP/IP or UDP sockets for any of the abovementioned protocols, ibaPDA would always bind the local sockets to all IP addresses configured on the machine (typically represented by the virtual IP address 0.0.0.0). This meant that e.g., when configuring the EGD interface for (UDP) port 18246 a socket was created for that port on all network adapters, even though EGD data might only be coming in through one network adapter. For security reasons and to avoid conflicts with other applications it is now possible to select for which network adapter(s) an interface should be configured.



For these TCP/IP and UDP-based interfaces a dropdown list has been added to their main interface node in the I/O manager where you can select for which network adapters the sockets should be opened. In case a network adapter has multiple IP addresses configured, a socket will be opened for all of these IP addresses.

Using the **All** and **None** options at the top of the dropdown list it is possible to quickly select or deselect all network adapters respectively. Note that for each of these interfaces, at least one

network adapter must be configured; if not, an error message will be produced when validating the I/O configuration.

For backwards compatibility, by default the option **All** is selected.

Note that when using an I/O configuration between different ibaPDA systems, the list of selected network adapters might not always be adapted completely. In case **All** is selected, all network adapters will be selected automatically on any system, regardless of the specific hardware. However, in case only specific network adapters are selected on e.g. system A, these adapters might not be available on system B. For each of the adapters configured on system A, ibaPDA will try to look for a best match on system B but in case no match is found, the network adapter will not be selected. A match is searched for as follows:

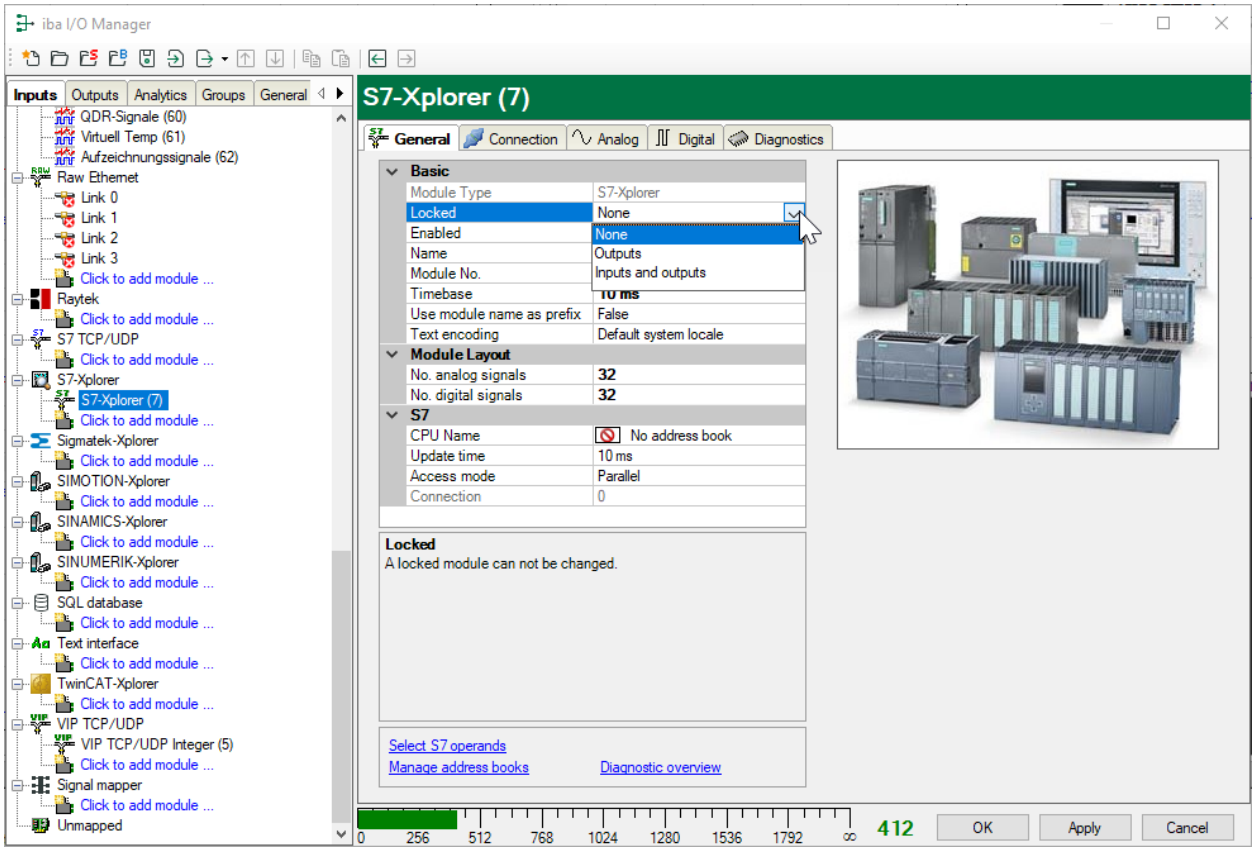
1. Complete name: [adapter name] ([hardware description]), e.g. LAN (Network adapter card 1235)
2. Partial adapter name. For example: if the adapter name in the configuration (from system A) is “LAN” and system B has adapters called “Local Area Network”, “LAN 2” and “Aux”, “LAN 2” will be considered a match.

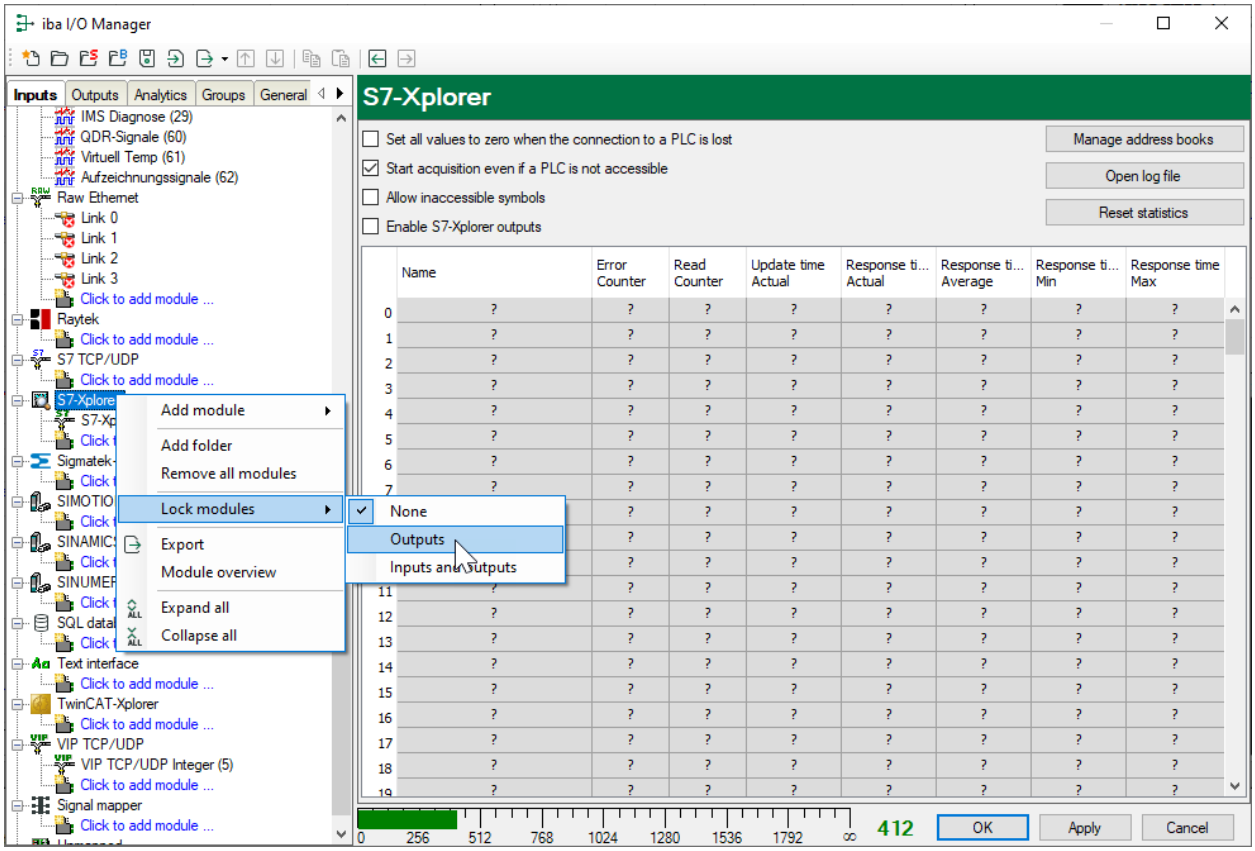
An additional change to these interfaces is that, when changing the port(s) for a certain interface a driver reload is no longer required, resulting in faster application of a new I/O configuration as well as more efficient memory handling in the ibaPDA driver.

3 Locking improvements

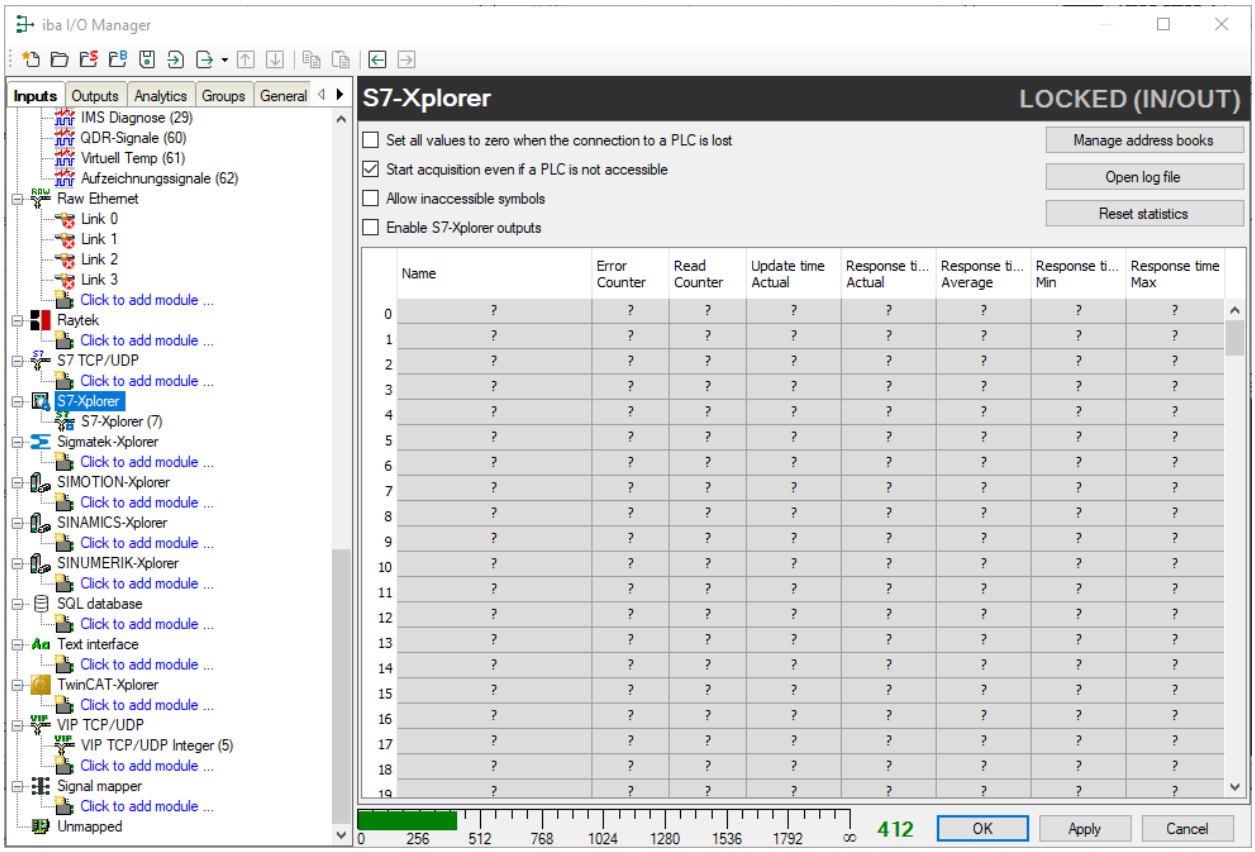
In the I/O manager you can lock modules. The configuration of locked modules cannot be changed. Only people with the right to “Change locked modules” can unlock a module and then make changes to its configuration. In ibaPDA versions before 8.6.0 a module could be locked or unlocked. Now in version 8.6.0 the locked property can have 3 possible values:

1. None: The module is unlocked. Changes can be made to the configuration.
2. Outputs: The output signals are locked and cannot be changed. The input signals can be changed. If there are module properties that affect the outputs then these are locked as well. This mode is new in ibaPDA v8.6.0.
3. Inputs and outputs: The complete module is locked. There can be no changes made to the module properties, the input signals and the output signals.





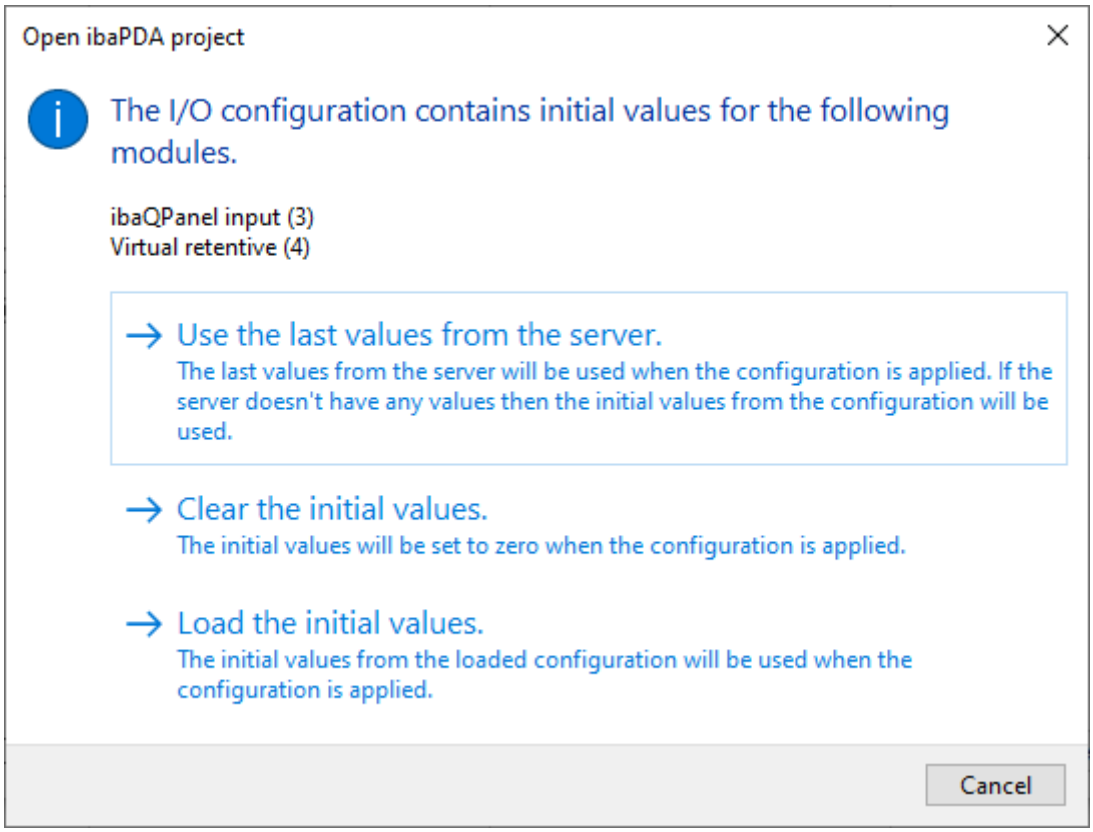
You can also lock all modules on an interface. This is done via right-clicking on the interface node.



When an interface is locked then no new modules can be added to it. But the properties of the interface can still be changed.

4 Loading project with initial values

There are a few modules that can store the last value of their signals when the acquisition is stopped. Two examples are the retentive virtual module and the ibaQPanel input module. This allows the signal values to be reused when the acquisition is started again. The initial values are stored in the I/O configuration. When loading a project that contains an I/O configuration with initial values for some modules then a dialog will open where the user can decide how to handle the initial values.



In ibaPDA versions before 8.6.0 only two options existed:

- Clear the initial values
- Load the initial values

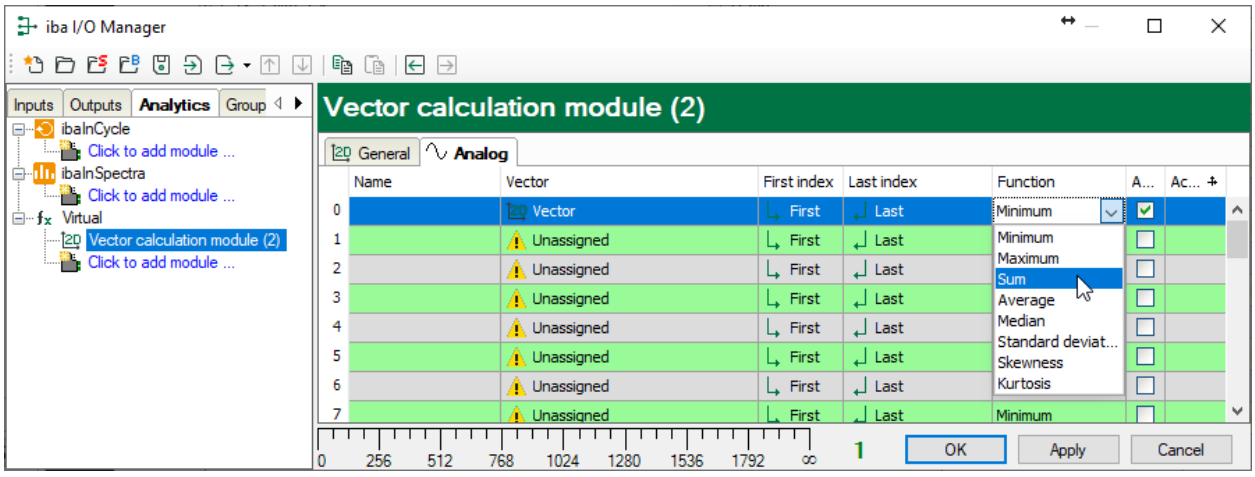
In ibaPDA v8.6.0 a third option has been added: Use the last values from the server. When the user selects that option then the last values of the currently running configuration will be used and if a corresponding module can't be found then the last values from the loaded project will be used.

5 Vector calculation module

The vector calculation module is a new module on the virtual interface that allows doing calculations on vectors. Vectors are multidimensional signals (arrays). The following functions are supported:

- Minimum
- Maximum
- Sum
- Average
- Median
- Standard deviation
- Skewness
- Kurtosis

The user can configure the vector and the required function for each signal in the analog tab of the module:



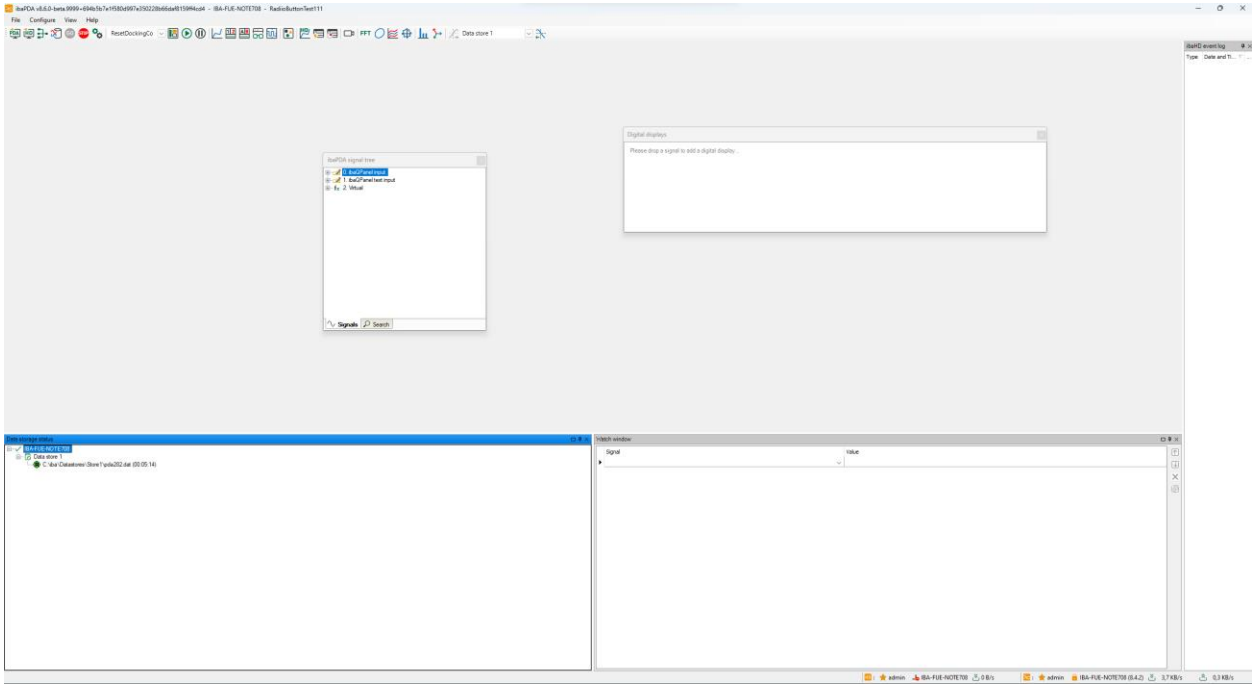
By default, all signals in the chosen vector are used for the calculation. It is also possible to process only a part of a vector. This can be done by configuring the first and the last index. These indices can be static, e.g. first index = 5, last index = 20. But the indices can also be configured to depend on a signal. This way, the exact vector signals taken into account for a calculation can change over time.

The timebase in the general tab of the module determines the update time of the calculations. Note that using smaller timebases in combination with large vectors and many different functions can cause a high load on the ibaPDA server.

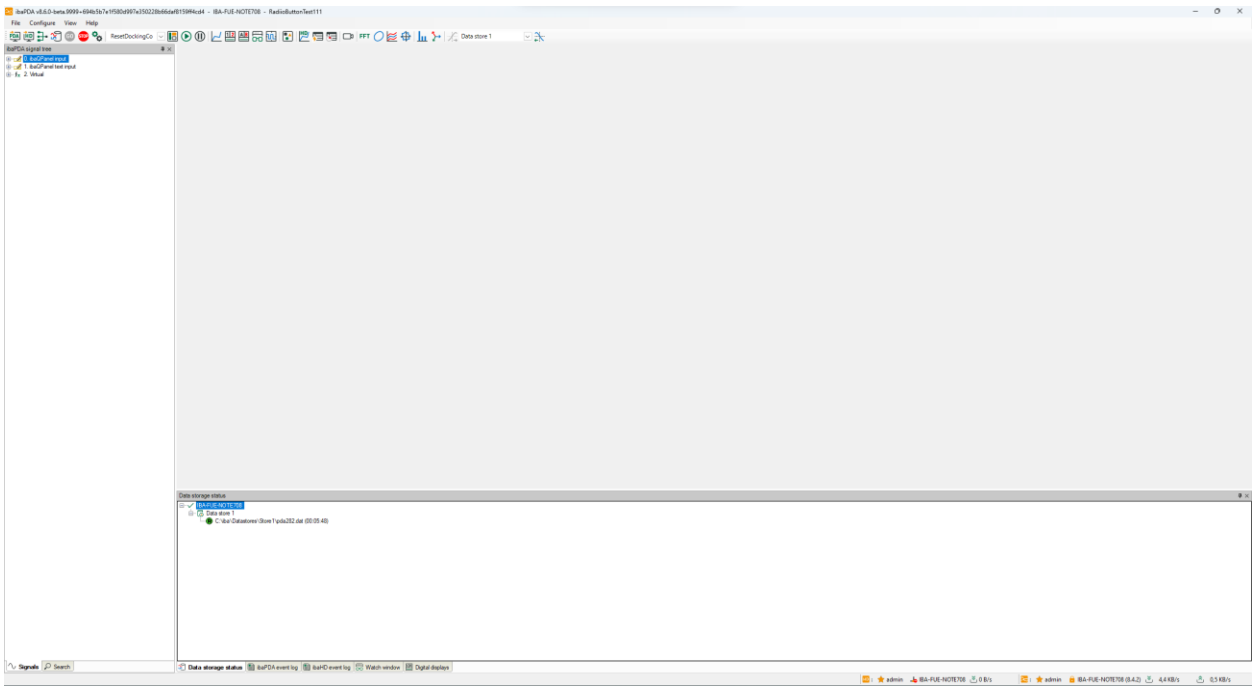
6 Reset docking configuration

The docking configuration of layouts can now be reset via new options in the view menu and layout manager. Resetting the docking configuration of a layout means that all of the windows within that layout (i.e. ibaPDA signal tree, Data storage status, etc.) are reset to their default visibility state, default size and docked to their default location.

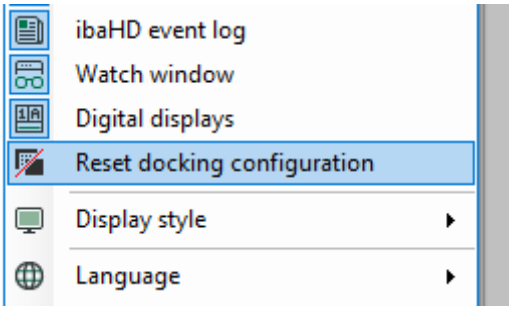
Before resetting docking configuration of the currently active layout:



After resetting docking configuration of the currently active layout:



The reset docking configuration option in the view menu resets the docking configuration of the currently active layout.



In the “Layouts” sub menu of the layout manager you can select the docking configurations of selected layouts, or the docking configurations of all layouts.