New Features in ibaDatCoordinator 2.0.0

Authors: Michael Verschaeve

Roman Kolesnik

Thomas George

Date: 7 September 2017

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

Table of contents

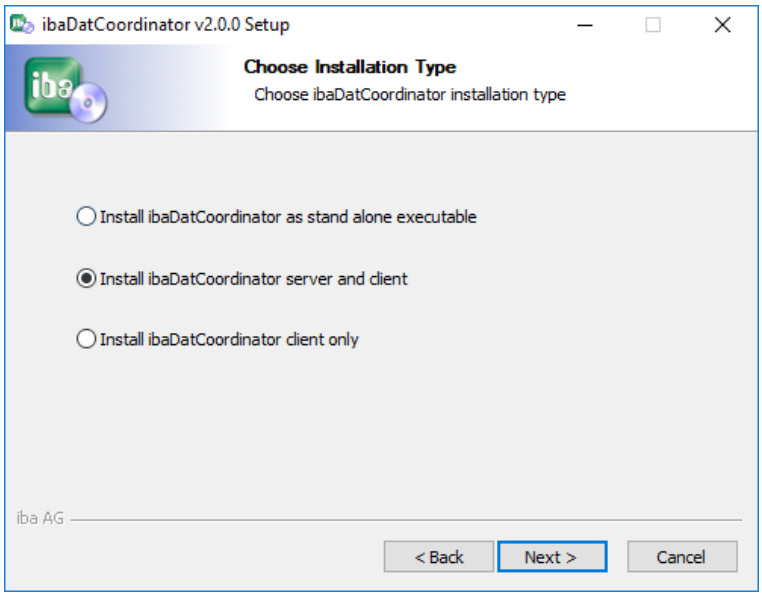
1 Client Server architecture

Besides installing ibaDatCoordinator as a standalone program, one can also install ibaDatCoordinator as a service with a client program. In previous versions of ibaDatCoordinator, this service needed to be on the same physical system. In the current version, this service can also be located and run on any remote system the client system has network access to.

1.1 Installing

When installing ibaDatCoordinator, the following types of installation are available.

- *Install ibaDatCoordinator as standalone executable*; this way ibaDatCoordinator will not run as a service, it runs locally in a single process.
- *Install ibaDatCoordinator server and client*; this will install both the client and server. Such installation can be used to run ibaDatCoordinator like in previous versions as a local service or one can use the installed service and connect to it from a client on another remote system.
- *Install ibaDatCoordinator client only*; use this installation type if you don't wish to run any instance of the ibaDatCoordinator service on the system but only wish to connect to remote ibaDatCoordinator servers.

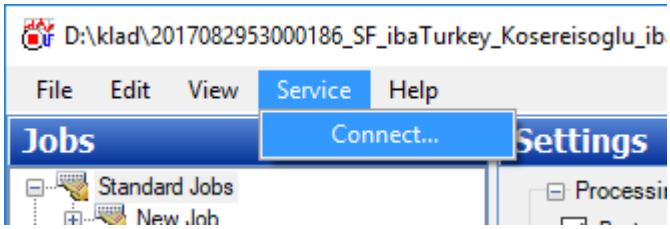


1.1.1 Plugins

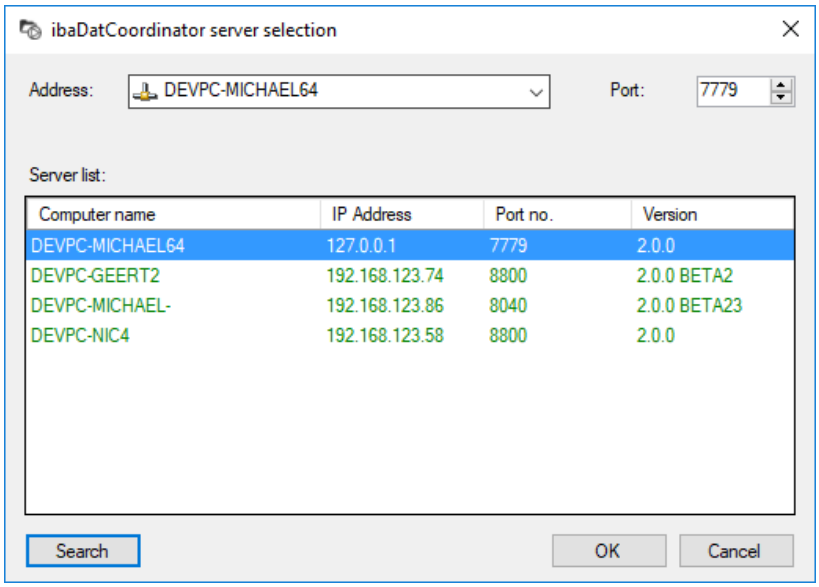
New versions of the ibaDatCoordinator plugins have been built to work with ibaDatCoordinator 2.0.0. The plugins need only be installed on the server system. When trying to install the plugins on a system that has only the client installed, an appropriate warning will be given and the installer will abort. When a client connects to a server for the first time, the necessary files to run the GUI components will be transferred from the server to the client. This will also happen when there is a discrepancy between the plugin versions. It's possible that the client will have already loaded plugin components with a different version before connecting to the server when it was connected to another server. In this case, the files cannot be copied; ibaDatCoordinator will ask if it can restart itself in order to properly copy the plugins.

1.2 Connecting to the server

On start, the ibaDatCoordinator client will try to connect automatically to the server it was last connected to. Initially this will be the local machine. You can select a different server by selecting from the main menu, the submenu *Service* and then *Connect*…



This will open the connection dialog:

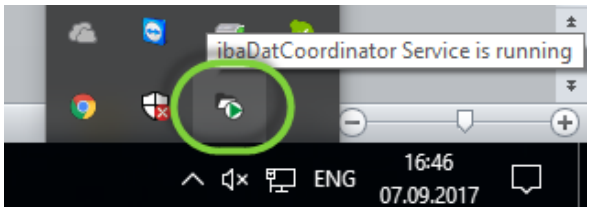


The connection dialog consists out of the following components:

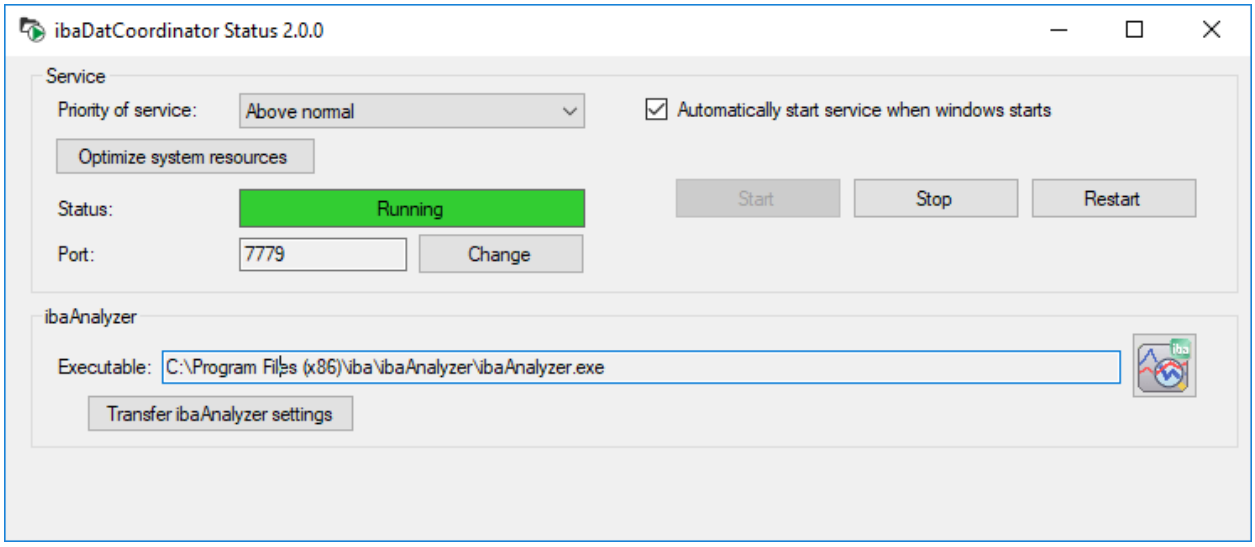
- You can type the hostname of the ibaDatCoordinator server in the textbox labeled "*Address*"
- This text box also contains a dropdown list where you can select one of the 10 most recent servers the client has connected to.
- You also need to specify the port number the client will use to communicate with the server through TCP/IP.
- Additionally there is a server list where you can select one of the servers that are currently found on the network.
- You can refresh this list by clicking the <*Search*> button.
- You can acknowledge the dialog by pressing the <*OK*> button, after which the client will try to connect to the specified server on the specified port.
- You can dismiss the dialog by clicking the <*Cancel*> button and stay disconnected or connected to the previously connected server.

1.3 Service Status

On systems where the ibaDatCoordinator service is running, a status program is automatically started with the status window initially minimized. You can open the status window by double clicking the system tray icon.



1.3.1 Status window

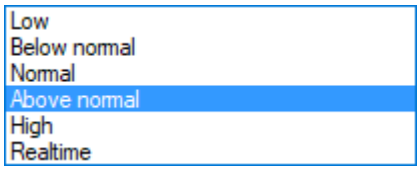


The status window consists out of two groups of options:

- The "*Service*" group.
- The "*ibaAnalyzer*" group.

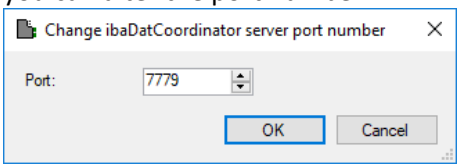
The Service group contains some options to control the service:

- You can set the priority of the service by selecting it from the dropdown box labeled so. Modifying the priority of the service will require a restart of the ibaDatCoordinator service. The status program will ask if it's ok to do so immediately or when the service is next started or restarted.



- You can also select whether or not the service starts automatically when Windows starts.

- A button labeled <*Optimize system resources*> is available; this button is useful for altering some system settings if the ibaDatCoordinator configuration contains a large number of tasks requiring ibaAnalyzer.
- The status of the service is displayed, “Running” in green or “Stopped” in red.
- Buttons are available to start, stop or restart the service.
- The port number is displayed which the clients must use to connect to this server through TCP/IP. The default ibaDatCoordinator port number is 8800.
- A button labeled <Change> is available next to the port number that will open a dialog where you can alter the port number.



The ibaAnalyzer group contains some options to control how the ibaDatCoordinator uses ibaAnalyzer:

- The location of the ibaAnalyzer executable is displayed, contrary to previous versions of ibaDatCoordinator where you could alter the ibaAnalyzer executable that was used, the textbox is currently read-only. We do no longer support changing the ibaAnalyzer executable from ibaDatCoordinator as theoretically only one ibaAnalyzer can be installed on a system.
- A button is available where you can start ibaAnalyzer.
- A button <*Transfer ibaAnalyzer settings*> is available, this will transfer the ibaAnalyzer settings of the current user to the user the service is running under (by default the system account).

1.3.2 System tray icon

The system tray icon will have a green arrow glyph when the service is running and will display the text “ibaDatCoordinator Service is running” as tooltip when the service is properly running.

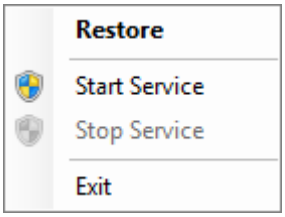
It will have a red block glyph if the service is stopped and display the text “ibaDatCoordinator Service is stopped” as tooltip.

If for some reason the service cannot be found (e.g. it was removed manually) a red X glyph will be present and the text “ibaDatCoordinator Service is not available” will be displayed as tooltip.

1.3.3 Context menu

When right-clicking on the system tray icon, a couple of shortcut options are available:

- Restore: this will open the status window, this is an alternative to double clicking the system tray icon.
- Start Service: Starts the service (grayed out when the service is already started)
- Stop Service: Stops the service (grayed out when the service is not started)
- Exit: closes the status program. Closing the status program, the system tray icon will also disappear. This is contrary to closing the status window, in which case the system tray icon still stays present.

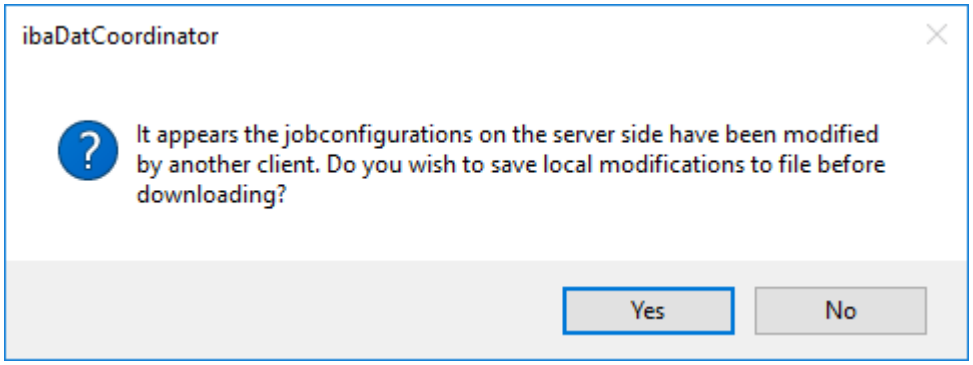


1.4 Multiple client connections

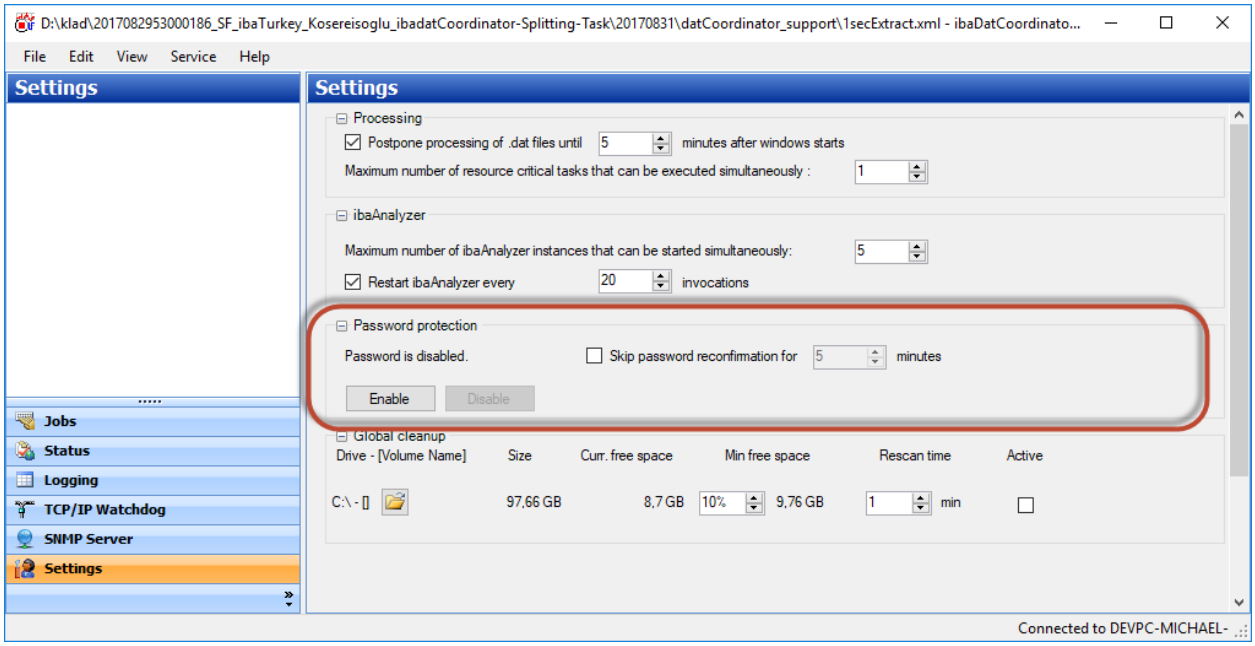
It is possible to connect from multiple clients to the same ibaDatCoordinator server.

This means that both can modify the ibaDatCoordinator jobs and tasks on the server.

Clients will detect if another client has made modifications and show a dialog where one has the option to save any local modifications before the changes from the other client are applied to the current client.



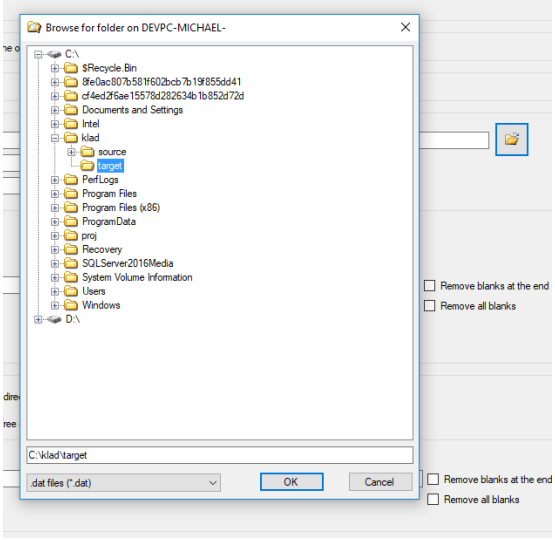
Note that if it is undesirable that multiple clients can change the ibaDatCoordinator configuration on the server, one can use the password functionality (already existing in previous ibaDatCoordinator versions) in the settings pane:


D:\klad\2017082953000186_SF_ibaTurkey_Kosereisoglu_ibadatCoordinator-Splitting-Task\20170831\datCoordinator_support\1secExtract.xml - ibaDatCoordinato...
File
Edit
View
Service
Help
Settings
Settings
Processing
Postpone processing of .dat files until
5
minutes after windows starts
Maximum number of resource critical tasks that can be executed simultaneously :
1
ibaAnalyzer
Maximum number of ibaAnalyzer instances that can be started simultaneously:
5
Restart ibaAnalyzer every
20
invocations
Password protection
Password is disabled.
Skip password reconfirmation for
5
minutes
Enable
Disable
Global cleanup
Drive - [Volume Name]
Size
Curr. free space
Min free space
Rescan time
Active
C:\ - []
97,66 GB
8,7 GB
10%
9,76 GB
1
min
Jobs
Status
Logging
TCP/IP Watchdog
SNMP Server
Settings
Connected to DEVPC-MICHAEL-

1.5 Changes to ibaDatCoordinator configuration

The GUI has remained largely the same and when installed as a standalone program or when the service is located on the same system, there are no notable changes compared to previous versions. However, some changes when running the service on a remote system:

- Analysis files, scripts, source and target directories are expected to be located on the remote system or to be located in a path accessible on the network; in the latter case it is advisable to specify an UNC path. When specifying a local filesystem path, it is expected to be a local path on the server system, if it is a path to a shared network folder, ibaDatCoordinator will translate the path to an UNC path. Browse buttons next to locations where you can specify such paths, will open a dialog that emulates a browse dialog running on the server system.



- The script task will load the text from a file on the remote server, and store the text on the remote server when saved.
- Options to start and stop the service have been moved from the client to the status program.
- Likewise, transferring ibaAnalyzer settings from the current logged in user to the service user happen in the status program.
- The client does not have a notify icon, you can close the client without it minimizing first.
- When generating a support file for iba support, the .zip file itself will contain two separate .zip files client.zip and server.zip. The latter was generated on the server side and transferred to the client side.

Some limitations are also currently present when running the service on a remote system:

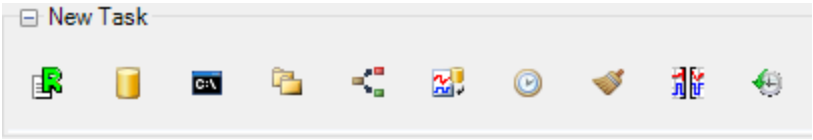
- For tasks where you can specify an analysis (located on the remote system or on a network share) it is currently not possible to start an ibaAnalyzer that loads the analysis.
- Testing the splitter task (see paragraph 2.1.2) is not supported.

When trying these actions on a client for which the service is on a remote system, an appropriate error message will be given; it is recommended to perform these actions on a client on the remote system.

2 New tasks

2.1 New splitter task

The ibaDatCoordinator is providing the new *Splitter* task



The Splitter task allows a *.dat* file to be split into several smaller *.dat* files which contain the same signals as the original *.dat* file, but with different (smaller) time ranges.

The task has three groups of options in common with any other task:

- “ID”: Here you can specify another name for the task than the default “Split”.
- “Execute”: Here you can specify when to execute the task.
- “Notify”: Here you can specify when ibaDatCoordinator should send notifications about the status of the task.

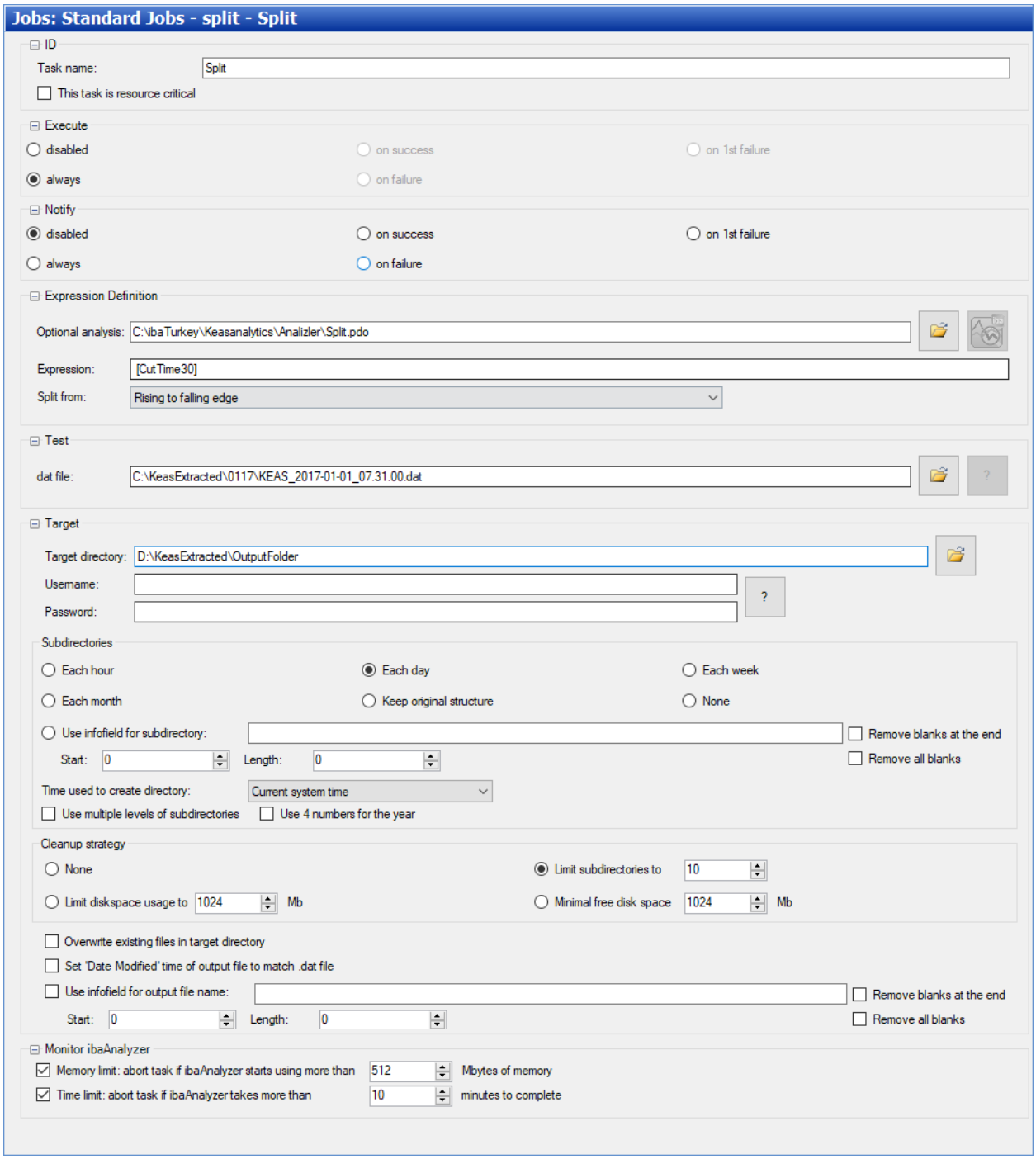
These options are entirely identical to the same options in any other task; consult the ibaDatCoordinator manual on how to configure them.

After the common options two additional groups of options are present:

- “Expression Definition”: Here you can specify how the input .dat file will be split into .dat files with smaller time ranges.
- “Test”: Here you can specify a test .dat file and perform a dry run on that .dat file to see how the specified .dat file will be split. Eventually you can split it.

Finally two groups are present that are again in common with other tasks:

- “Target”: Here you specify the target directory for the resulting .dat files, along with several other options. These options are entirely identical to the same options in other tasks that produce output in a file folder; consult the ibaDatCoordinator manual on how to configure them.
- “Monitor ibaAnalyzer”: The splitter task uses ibaAnalyzer to interpret the criterion to split the .dat file. In this group of options you can specify a memory limit on the amount of memory ibaAnalyzer is allowed to use to evaluate the criterion, as well as a time limit. These options are identical to the same options in other tasks that use ibaAnalyzer, consult the ibaDatCoordinator manual for further information.

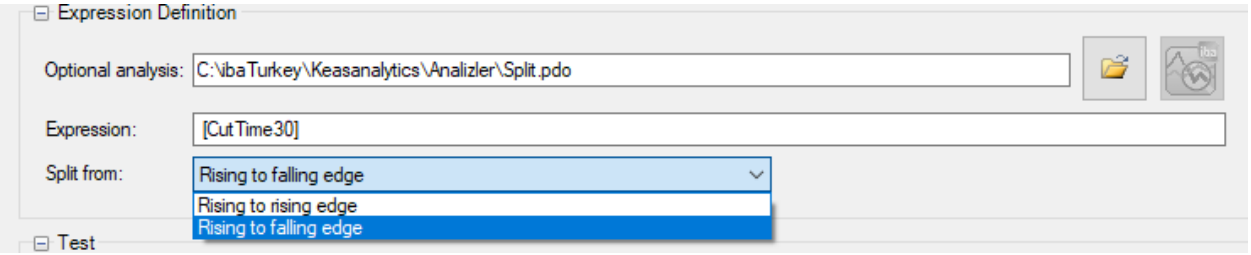

Jobs: Standard Jobs - split - Split
ID
Task name:
Split
This task is resource critical
Execute
disabled
always
on success
on failure
on 1st failure
Notify
disabled
always
on success
on failure
on 1st failure
Expression Definition
Optional analysis:
C:\ibaTurkey\Keasanalytics\Analizler\Split.pdo
Expression:
[CutTime30]
Split from:
Rising to falling edge
Test
dat file:
C:\KeasExtracted\0117\KEAS_2017-01-01_07.31.00.dat
Target
Target directory:
D:\KeasExtracted\OutputFolder
Username:
Password:
Subdirectories
Each hour
Each day
Each week
Each month
Keep original structure
None
Use infofield for subdirectory:
Remove blanks at the end
Start:
0
Length:
0
Remove all blanks
Time used to create directory:
Current system time
Use multiple levels of subdirectories
Use 4 numbers for the year
Cleanup strategy
None
Limit subdirectories to
10
Limit diskspace usage to
1024
Mb
Minimal free disk space
1024
Mb
Overwrite existing files in target directory
Set 'Date Modified' time of output file to match .dat file
Use infofield for output file name:
Remove blanks at the end
Start:
0
Length:
0
Remove all blanks
Monitor ibaAnalyzer
Memory limit: abort task if ibaAnalyzer starts using more than
512
Mbytes of memory
Time limit: abort task if ibaAnalyzer takes more than
10
minutes to complete

2.1.1 Expression Definition

The expression group consists out of the following options

- A textbox labeled “Optional analysis” where you can type the path to an analysis; there is also a browse button that opens a file browser that allows you to browse for the analysis (.pdo file). Additionally there is a button that when a valid path is entered into the textbox, you can click to open the analysis in ibaAnalyzer. Specifying an analysis is optional, if the expression that determines how the .dat file needs to be split (e.g. a single signal in the .dat file) is simple enough, specifying an analysis can be omitted.
- A textbox labeled “Expression” where you type the expression to be used to split the .dat file. Any valid ibaAnalyzer expression is allowed. Also if an analysis is specified, the expression can refer to other expressions specified in the analysis. Expression is expected to be a digital signal evaluated in the time domain.
- Finally a selection box labeled “Split from” is present where you can decide how the signal specified by “Expression” will be interpreted to split the .dat files.
    - If one selects the option “Rising to rising edge”, the *.dat* file will be split from the start of the *.dat* file time range to the first transition of “*Expression*” from false to true. Then it will be further split from any false/true transition to the next subsequent false/true transition. Finally the file will be split from the last false/true transition until the end of the *.dat* file time range.
    - If one selects the option “*Rising to falling edge*” only the parts of the *.dat* file time range where “*Expression*” evaluates to true are retained. For each block where “*Expression*” was true, a separate file is generated.

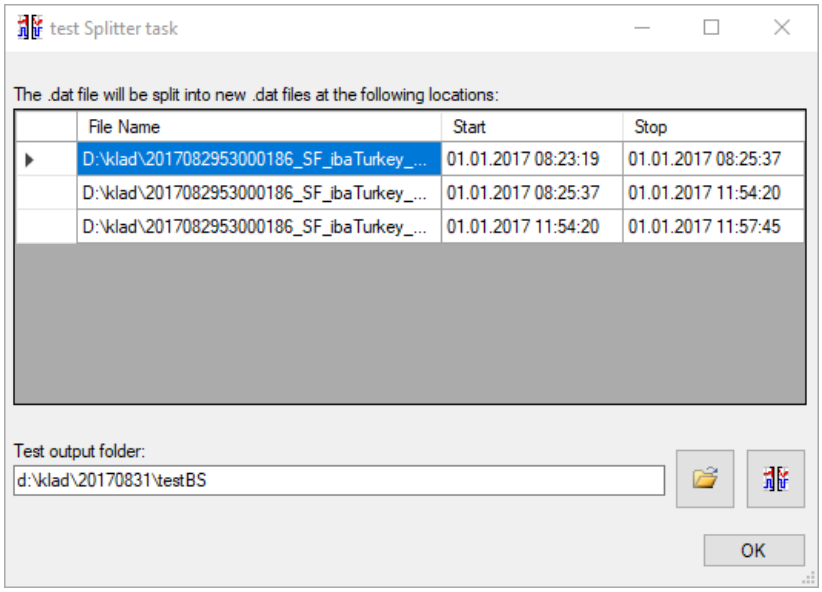
Note that when opening the resulting files in ibaAnalyzer and appending them to each other, and the option “*Rising to rising edge*” was selected the entire original file will be reproduced, no data is lost. When doing so when “*Rising to falling edge*” was selected, gaps in the signals will appear where “*Expression*” was false, this data was not retained.



2.1.2 Test options and test dialog

In the group labeled *"Test"* one can specify a .dat file that can be used to test the splitting. A textbox labeled "*dat file*" is present where you can type the path to a .dat file. Additionally a browse button is available that opens a file browser dialog to browse for the test .dat file.

A button with a question mark is present which you can click if you have specified a valid path to the test .dat file. Clicking the button opens the test dialog.



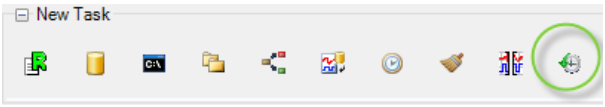
When opening the dialog the grid will start filling up with candidate split files. Depending on the complexity of the split expression, optional analysis and the amount of data, this might take a while. You can abort the process by clicking the stop button.

You can specify a path in the textbox labeled "Test output folder" to a directory where you want the resulting .dat files generated. Additionally there is a browse button to open a browse dialog to browse for a folder. Finally, there is the <Split> button that will extract the .dat files with the calculated subranges in the specified folder. The <Split> button will again turn into a stop button if you want to abort this process.

You can dismiss the dialog by pressing the <OK> button.

2.2 Import iba data files to iba Historical Data server

The ibaDatCoordinator is providing the new task *ibaHD import*



The task provides the function to import iba data files to iba Historical Data server.

After adding an ibaHD import task to your job, you can configure the destination for the dat file data by selecting your ibaHD Server and one HD store. You have the option to limit the transfer rate of the data. You can create new text channels from info fields of your dat files.

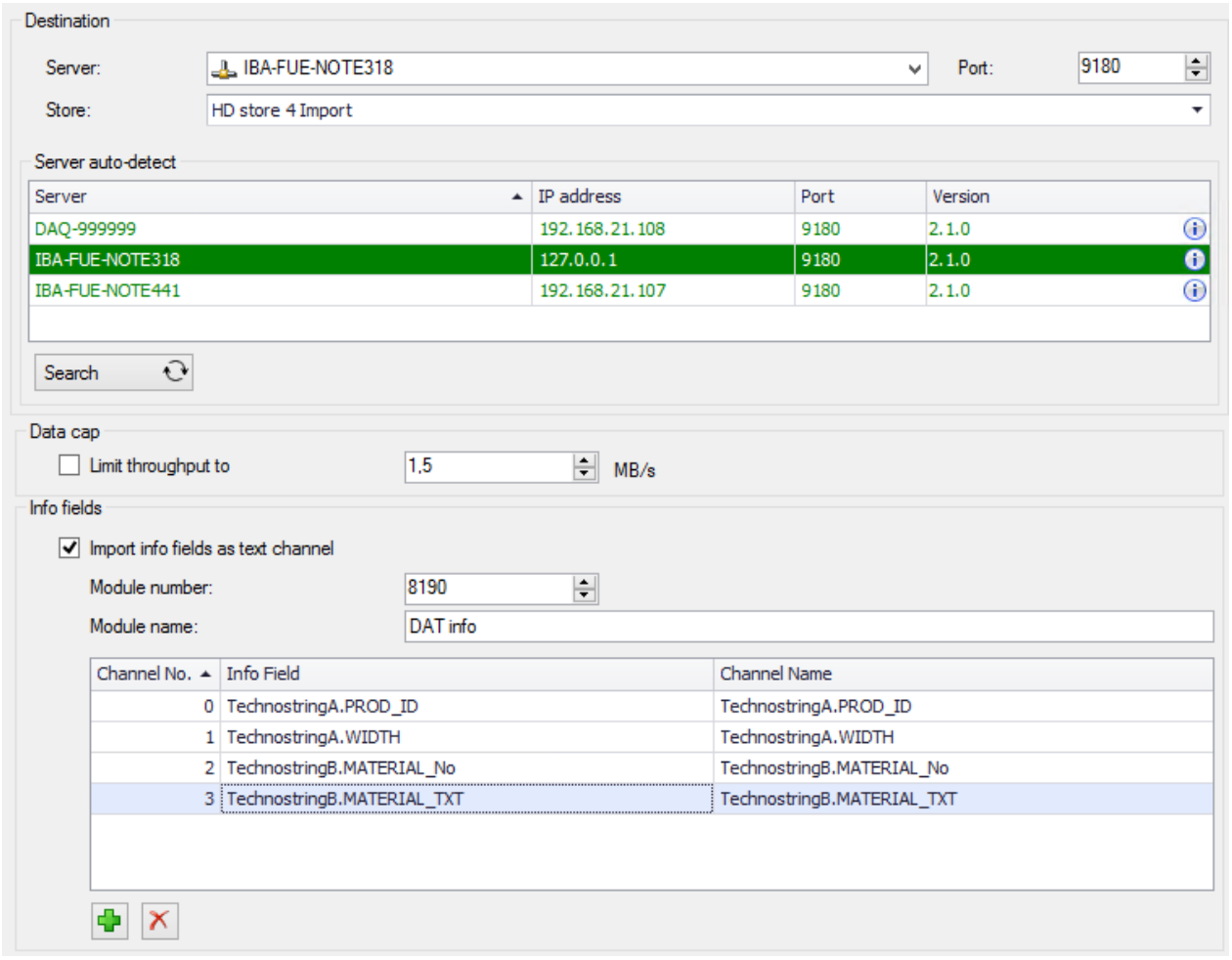
The task has three groups of options in common with any other task:

- *"ID"*: Here you can specify another name for the task than the default "ibaHD import".
- *"Execute":* Here you can specify when to execute the task.
- *"Notify":* Here you can specify when ibaDatCoordinator should send notifications about the status of the task.

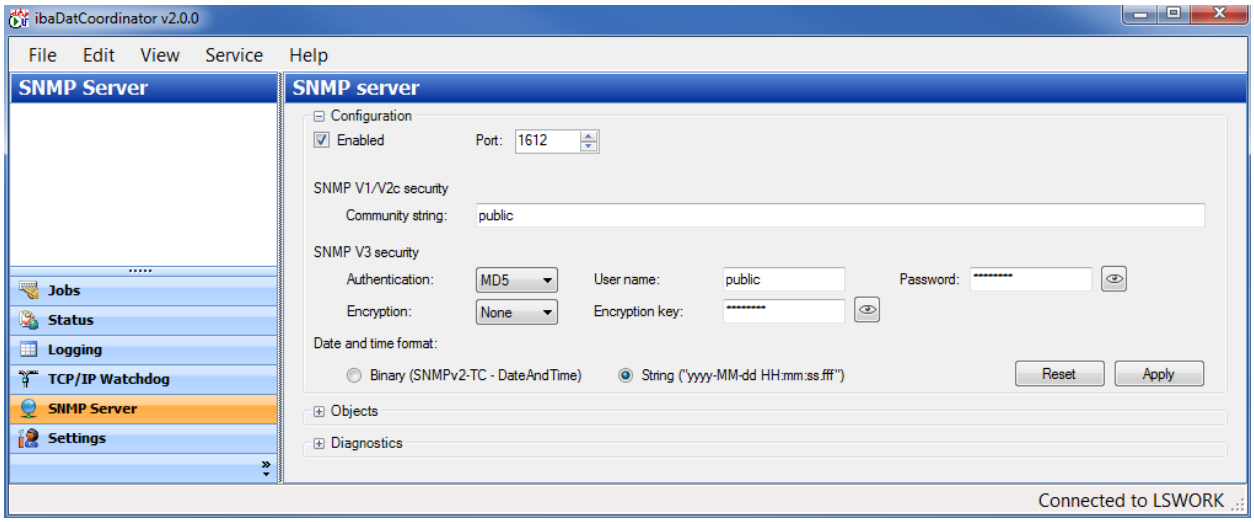
These options are entirely identical to the same options in any other task; consult the ibaDatCoordinator manual on how to configure them.

After the common options the specific *ibaHD import* options are presented in a tab window consisting out of several tabs:

- "Destination": here you can select or type directly the IP address or the name of HD Server and the time based HD store as destination of the iba file data.
- "Data Cap": The transfer rate can be limited. This might be important in case the destination store has a concurrent online acquisition.
- "Info fields": Specify info fields that will be read from the data files and will be converted into text channels of the module with specified number and name.


Destination
Server:
IBA-FUE-NOTE318
Port:
9180
Store:
HD store 4 Import
Server auto-detect
Server
IP address
Port
Version
DAQ-999999
192.168.21.108
9180
2.1.0
IBA-FUE-NOTE318
127.0.0.1
9180
2.1.0
IBA-FUE-NOTE441
192.168.21.107
9180
2.1.0
Search
Data cap
Limit throughput to
1,5
MB/s
Info fields
Import info fields as text channel
Module number:
8190
Module name:
DAT info
Channel No.
Info Field
Channel Name
0
TechnostringA.PROD_ID
TechnostringA.PROD_ID
1
TechnostringA.WIDTH
TechnostringA.WIDTH
2
TechnostringB.MATERIAL_No
TechnostringB.MATERIAL_No
3
TechnostringB.MATERIAL_TXT
TechnostringB.MATERIAL_TXT

3 SNMP server

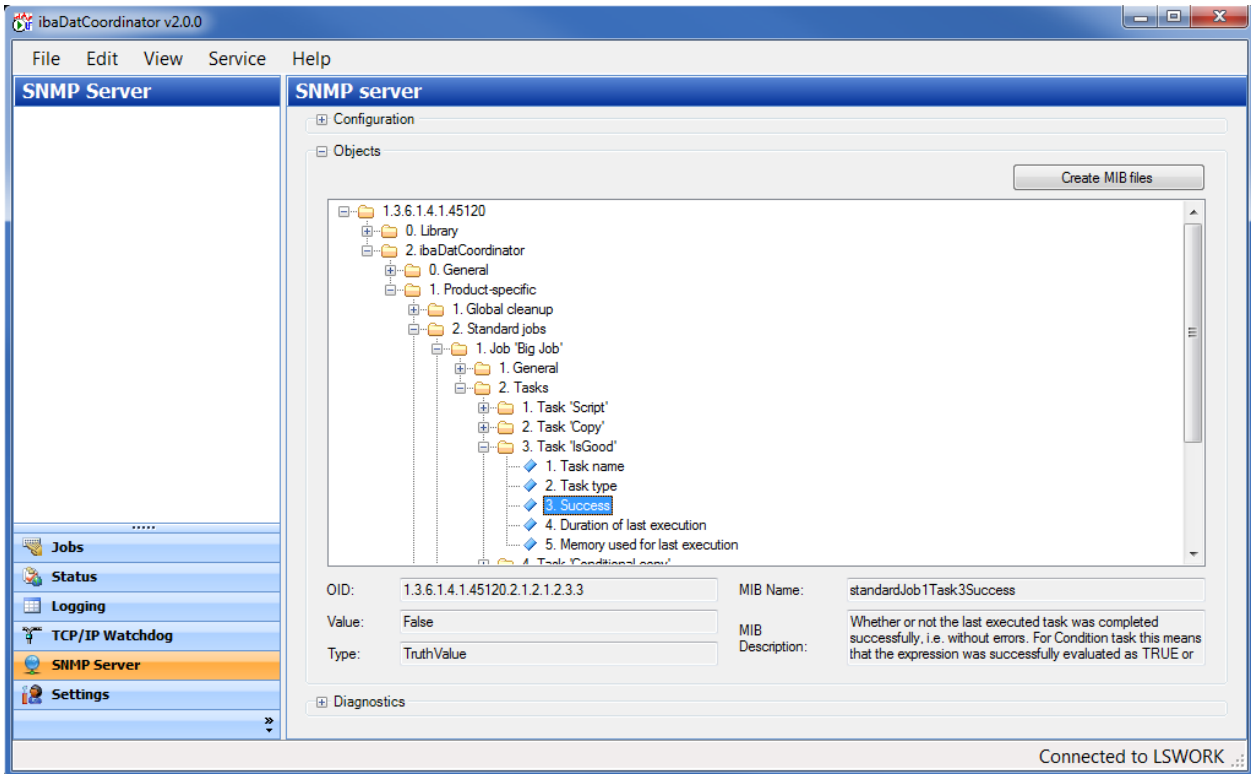


ibaDatCoordinator is now an SNMP server. By default it is disabled. You can enable it and set it up on the "*SNMP Server*" pane in the "*Configuration*" section. You can configure the port where the SNMP server is available on. ibaDatCoordinator supports the SNMP protocols V1, V2c and V3. For V1 and V2c you can only provide a community string that acts as a password. The default is "public" so that the SNMP server is accessible by any SNMP client. For V3 the security settings are a bit more elaborate. You can configure authentication and encryption.

Additionally, you can set the format of date/time values: Binary (encoded into several bytes according to SNMPv2-TC standard) or String (Human-readable format).

Click <*Apply*> button to apply your settings immediately. Otherwise settings will be applied automatically when you leave the "*SNMP Server*" pane. Click <*Reset*> button to reset all the settings (except Enabled/Disabled) to defaults.

You do not need any additional license to use the SNMP Server.



In the "*Objects"* section you can see which data is available on the SNMP server. The tree shows the OID (object identifier) tree. All the objects that you see here will be available over SNMP.

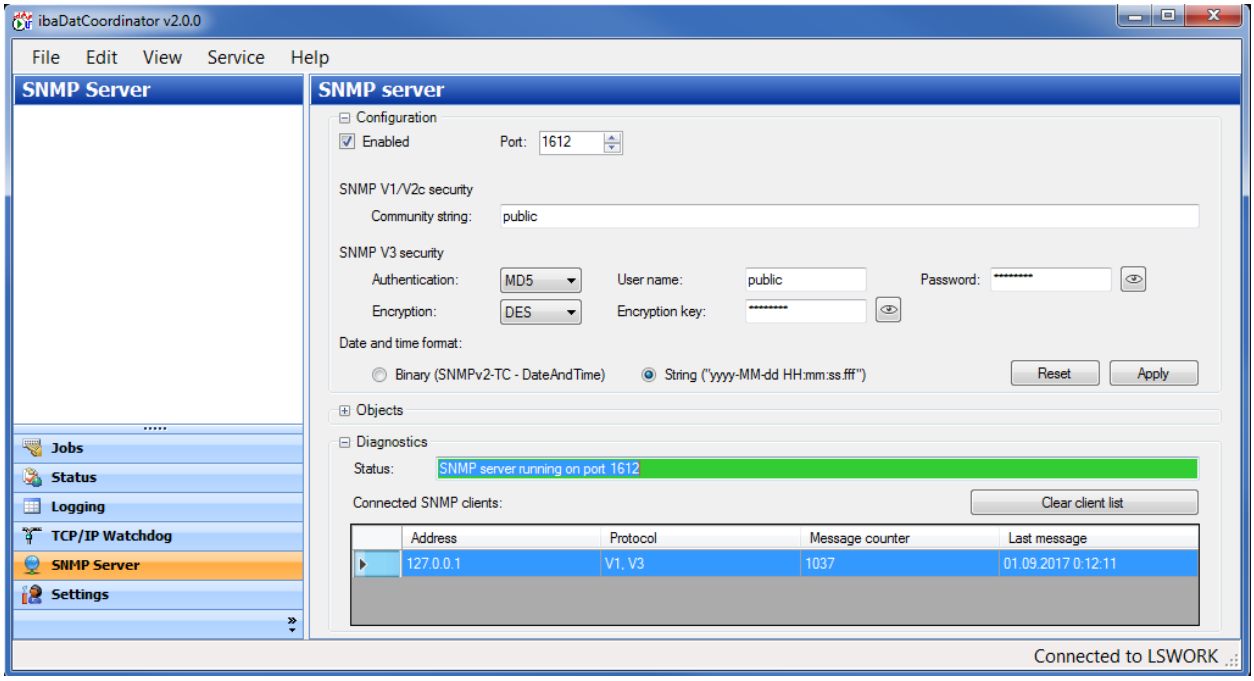
All values are divided into several groups:

- Objects about the **SNMP library**
- Objects about **ibaDatCoordinator** general properties like the version and dongle information
- Objects about **Global Cleanup**
- Objects about **Standard Jobs**
- Objects about **Scheduled Jobs**
- Objects about **One time Jobs**

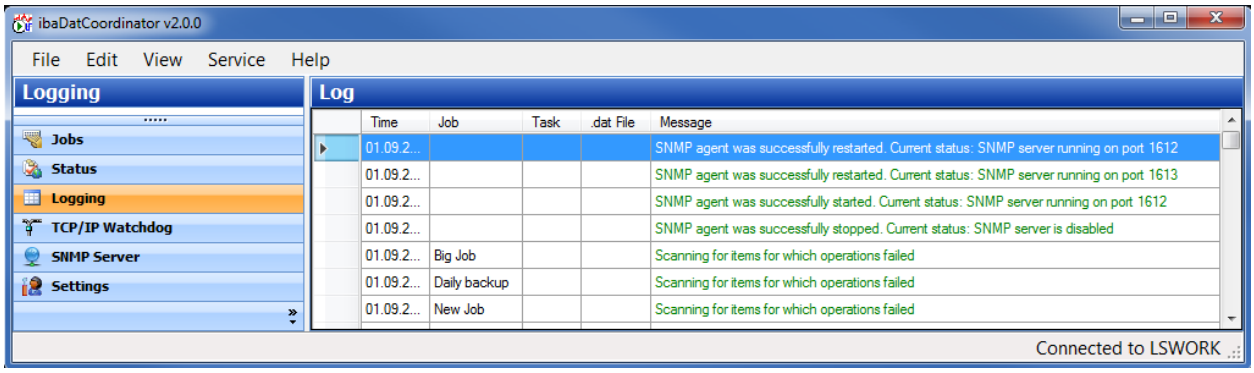
On the top there is also a button to create the corresponding MIB files. The MIB files are generated by the ibaDatCoordinator server. So, make sure you've applied your local job settings (see "Jobs" Pane) to the ibaDatCoordinator server before generating MIB files (otherwise, MIB files will not contain your most recent local changes). Anyway, you can preview current OID structure using the tree control. MIB files will exactly correspond to what you see in this tree. There are 2 MIB files generated:

- IBA-GENERAL-MIB.txt: This contains the objects that are common to all iba products.
- IBA-PRODUCT-IBADATCOORDINATOR-MIB.txt: This contains the ibaDatCoordinator specific objects.

Below the OID tree you can see the full OID, the MIB name and the MIB description of the selected node. You can also see the current value and the data type of the selected node.



In the "*Diagnostics*" section you can see the current status of the SNMP server. It also shows a list of connected SNMP clients. Only the clients that have accessed the SNMP server within the last hour are shown. Via the <*Clear client list*> button you can remove all the clients from the list.



Changes of SNMP server status are also reflected in the log. This occurs on server Start, Stop or on other changes in its "*Configuration*" section.