



User's manual

2022-11



@ 2022, TECMATIA SL

Industrial Park PPI-7, Parcel 1, Ground Floor, Office A

36475 - O Porriño (Pontevedra) - Spain

Tel: (+34) 986 11 63 18

E-mail: info@tecmatia.com

Web: www.tecmatia.com

All rights reserved.

The proper names of programs, operating systems, hardware equipment, etc., that appear in this manual are registered trademarks of their respective companies or organizations.

Despite the fact that Tecmatia has put all possible care and precaution in the development of the software, as well as in the preparation and updating of this manual, it does not assume responsibility for possible damages resulting from the use of the application or this document, or from errors or omissions in the information it may contain.

If this document is not enough to resolve your doubts, if you wish to report any problem related to the application, or request possible improvements or functionalities, do not hesitate to contact Tecmatia.

This manual is based on version 4.24.0. You may have a later version of the application, but this does not necessarily mean that the manual is out of date, as there may be revisions without new functionality. Check the web or ask Tecmatia or your distributor if a more current version of the manual is available.

Index

1	INTRODUCTION	9
2	INSTALLATION	10
2.1	SYSTEM REQUIREMENTS	10
2.2	INSTALLING THE APPLICATION	10
2.3	LICENSE	10
2.3.1	<i>Request a license</i>	10
2.3.2	<i>Activate the license</i>	11
2.4	UPDATE THE APP	12
3	STARTING THE APPLICATION	13
4	INTERFACE	14
5	SECURITY: USER CONTROL	15
6	VARIABLES	16
6.1	QUERY AND VISUALIZATION	17
6.2	DATA CLIENT CONFIGURATION	18
6.2.1	<i>Data source types</i>	19
6.2.1.1	API	19
6.2.1.2	Database	20
6.2.1.3	Calculated data	20
6.2.1.4	Internal data	21
6.2.1.5	Ethernet/IP	21
6.2.1.6	Historical in DB	21
6.2.1.7	Microcom Zeus	22
6.2.1.8	Modbus	22
6.2.1.9	OPC-DA	24
6.2.1.10	OPC-UA	25
6.2.1.11	Network	26
6.2.1.12	Siemens S7	26
6.2.1.13	File system	27
6.2.1.14	Operating system	27
6.2.1.15	Telegraf (InfluxData)	27
6.2.2	<i>Reading groups</i>	28
6.2.2.1	API	29
6.2.2.2	Database	29
6.2.2.3	Historical in DB	30
6.2.2.4	Microcom Zeus	30
6.2.2.5	Modbus	30
6.2.2.6	File system	31
6.2.2.7	Telegraf (InfluxData)	31
6.2.3	<i>Editing</i>	31
6.3	VARIABLES	32
6.3.1	<i>General</i>	32
6.3.2	<i>Display</i>	32
6.3.3	<i>Acquisition > Transformation</i>	33
6.3.4	<i>Acquisition > Value mapping</i>	36
6.3.5	<i>Transfer</i>	36
6.3.6	<i>Actions</i>	36
6.3.7	<i>Data log</i>	36
6.3.8	<i>Bindings</i>	37
6.3.9	<i>Data source types</i>	37
6.3.9.1	API	37
6.3.9.2	Database	37
6.3.9.3	Calculated data	38
6.3.9.4	Internal data	39

6.3.9.5	Ethernet/IP.....	39
6.3.9.6	Historical in DB.....	40
6.3.9.7	Microcom Zeus.....	40
6.3.9.8	Modbus.....	40
6.3.9.9	OPC.....	41
6.3.9.10	Network.....	41
6.3.9.11	Siemens S7.....	41
6.3.9.12	File system.....	42
6.3.9.13	OS.....	42
6.3.9.14	Telegraf (InfluxData).....	44
6.3.10	<i>Editing</i>	44
6.3.11	<i>One variable alarms</i>	45
6.3.12	<i>Simulate value</i>	45
7	ALARMS.....	46
7.1	SYSTEM ALARMS.....	46
7.2	PROJECT ALARMS.....	47
7.3	QUERY AND VISUALIZATION.....	47
7.3.1	<i>Summary</i>	47
7.3.2	<i>Historian</i>	48
7.4	EDITING AND CONFIGURATION.....	48
7.5	ACKNOWLEDGMENT OF ALARMS.....	52
7.6	MUTE ALARMS.....	52
7.7	SIMULATE ACTIVATION.....	52
8	NOTIFICATION POLICIES.....	53
9	QUERIES.....	55
9.1	PREDEFINED SYSTEM QUERIES.....	55
9.2	USER QUERIES.....	56
9.2.1	<i>Display</i>	56
9.2.2	<i>Editing</i>	56
9.2.3	<i>Automated queries</i>	56
9.3	DIRECT QUERY.....	57
10	ACTIONS.....	58
10.1	PREDEFINED SYSTEM ACTIONS.....	58
10.2	USER ACTIONS.....	59
10.2.1	<i>Writing</i>	59
10.2.2	<i>Execute application</i>	59
10.2.3	<i>Sound</i>	59
10.2.4	<i>Windows services</i>	60
10.2.5	<i>Scheduled Tasks</i>	60
10.2.6	<i>Send SMS / email</i>	60
10.2.7	<i>InfluxDB: Writing</i>	60
10.2.8	<i>InfluxDB: Execute</i>	60
10.3	DISPLAY.....	60
10.4	DIRECT WRITING OF A VARIABLE VIA MESSAGE.....	61
11	NOTIFICATION / MESSAGING SYSTEMS.....	62
11.1	OPERATION SCREEN.....	62
11.1.1	<i>Operate</i>	62
11.1.2	<i>Counters</i>	63
11.1.3	<i>Log</i>	63
11.2	E-MAIL.....	63
11.3	GSM MODEM (SMS).....	63
11.4	SMS ONLINE SERVICES (SMS GATEWAYS).....	64
11.5	EXTERNAL MESSAGES.....	65

11.6	MESSAGE HISTORY	65
11.7	PERIODIC STATUS REPORT	66
12	CONTACTS	67
12.1	DISPLAY	67
12.2	EDITING	68
12.2.1	<i>Teams</i>	68
12.2.2	<i>Contacts</i>	68
13	HISTORICAL DATA.....	71
13.1	INTERNAL DATA LOG	71
13.2	DATA LOG IN EXTERNAL DATABASE (SQL)	71
13.3	DATA LOG IN EXTERNAL DATABASE (INFLUXDB)	72
13.4	INTERNAL GRAPHS.....	73
13.4.1	<i>Selection panel</i>	73
13.4.2	<i>Graph and data panel</i>	74
13.4.3	<i>Statistics panel</i>	74
14	OPC-DA SERVER.....	75
15	API REST	77
15.1	BASE URL	77
15.2	API KEY	77
15.3	HTTP RESULTS	77
15.4	ENDPOINTS.....	77
15.4.1	<i>Alarms</i>	77
15.4.2	<i>Variables</i>	78
15.4.3	<i>Internos</i>	79
15.4.4	<i>Actions</i>	79
15.4.5	<i>Messages</i>	79
16	WEB ACCESS	80
17	SUBSTITUTION PARAMETERS	82
17.1	GENERIC DATA	82
17.2	PC DATA	82
17.3	APPLICATION DATA.....	82
17.4	VARIABLES DATA	83
17.5	ALARM / EVENT DATA	83
17.6	CONTACTS DATA	84
17.7	MESSAGE DATA.....	85
17.8	MODIFIERS.....	85
17.9	WHERE CAN THEY BE USED?.....	86
18	BACKUP COPY.....	87
18.1	BACKUP TYPES.....	87
18.2	AUTOMATIC BACKUP	87
18.3	MANUAL BACKUP	87
18.4	RESTORE A BACKUP	87
19	SETTINGS.....	89
19.1	GLOBAL	89
19.1.1	<i>Global</i>	89
19.1.2	<i>Project</i>	90
19.2	CONTROL PANEL.....	90
19.3	USERS	90
19.3.1	<i>Users</i>	90
19.3.2	<i>Permissions</i>	91

19.4	DATA.....	91
19.4.1	<i>Data source</i>	91
19.4.2	<i>OPC-DA server</i>	91
19.4.3	<i>REST API</i>	92
19.5	ALARMS.....	93
19.5.1	<i>Alarms</i>	93
19.5.2	<i>Notification</i>	93
19.5.3	<i>Message templates</i>	94
19.5.4	<i>Suffixes when linking alarm name</i>	95
19.6	MESSAGES	95
19.6.1	<i>Status report</i>	95
19.6.2	<i>Messages from external sources</i>	95
19.6.3	<i>Translation</i>	97
19.7	QUERIES / ACTIONS	97
19.8	SMS	98
19.8.1	<i>SMS</i>	98
19.8.2	<i>GSM modem</i>	98
19.8.3	<i>SMS gateway (online services)</i>	100
19.9	EMAIL.....	101
19.9.1	<i>Output (SMTP)</i>	101
19.9.2	<i>Input (POP3 / IMAP)</i>	102
19.10	DATA LOG	103
19.10.1	<i>Internal</i>	103
19.10.2	<i>SQL: Variables</i>	103
19.10.3	<i>SQL: Alarms</i>	106
19.10.4	<i>SQL: Custom fields</i>	107
19.10.5	<i>InfluxDB</i>	107
19.10.6	<i>FlyThings</i>	109
19.10.7	<i>Backup</i>	109
19.10.8	<i>Disk space used</i>	109
19.11	TRANSFORMATIONS	109
19.11.1	<i>Point curves</i>	109
19.11.2	<i>Timetables</i>	110
20	LOG FILES.....	112
21	EXPORT / IMPORT SETTINGS	113
22	KEYBOARD SHORTCUTS	114
23	FAQ / TROUBLESHOOTING	115

Illustration Index

Illustration 1: Start window and system tray icon.	13
Illustration 2: Notify main screen.....	14
Illustration 3: User login.....	15
Ilustración 4: Available data clients	16
Illustration 5: Variables section.....	17
Illustration 6: External API client settings.	19
Illustration 7: Database client settings.....	20
Illustration 8: Calculated data client settings.....	20
Illustration 9: Internal Data client settings.....	21
Illustration 10: Ethernet/IP client settings.	21
Illustration 11: Historical in DB client settings.	22
Illustration 12: Microcom Zeus client settings.....	22
Illustration13: Modbus client settings.	23
Illustration 14: OPC-DA client settings.....	24
Illustration 15: OPC-UA client settings.	25
Illustration 16: Network data client settings.....	26
Illustration 17: Siemens S7 client settings.	26
Illustration 18: File system data client settings.....	27
Illustration 19: Operating system data client settings.	27
Illustration 20: Telegraf client settings.....	28
Illustration 21: Reading group settings.	29
Illustration 22: Alarms section.	46
Illustration 23: Notification policies.	53
Illustration 24: GSM modem operation	62
Illustration 25: Message history.....	65
Illustration 26: Contacts.	67
Illustration 27: Timetable settings	70
Illustration 28: Historian in InfluxDB + Grafana	72
Illustration 29: Graph for one variable.....	73
Illustration 30: Grafana - Notify Dashboard Summary	80
Illustration 31: Grafana - Historical Notify Dashboard.....	81
Illustration 32: Grafana - Graphical Notify Dashboard	81
Illustration 33: Main Settings	89
Illustration 34: Editing a message template.....	94

Illustration 35: Example of HTML email template	94
Illustration 36: External database connection configuration.....	104
Illustration 37: Point curves.	110
Illustration 38: Periods timetable.	111
Illustration 39: ON/OFF timetable.....	111

1 Introduction

Notify is a software application that is part of the *NEXIA DM* software suite from Tecmatia. Its main function is the **notification and management of alarms** or alerts, from data collected from OPC or Modbus servers, directly from PLCs, databases, or data from the system itself, and its subsequent notification to the user by SMS or e-mail. However, as you will see along this manual, it has other powerful features, such as queries, writes, data logging, graphical and historical queries, statistics, etc.

Access to data via **OPC-DA¹**, **OPC-UA²**, **Modbus³**, **Ethernet/IP**, or **Siemens S7**, standard protocols widely used in the industrial automation sector, makes it possible to connect with almost any device in real time without the need for proprietary drivers. And not only industrial protocols, it also allows you to read from **databases**, **operating system**, **file system**, etc. Its **integrated OPC-DA server** and a REST API allows to subscribe third applications to Notify data that has been obtained from another protocol, or that has been processed inside the application, so Notify can also be used as a protocol gateway.

The separation of the concept of variables and alarms provides versatility and added value, being able to have the base data on one side, be it for example a flow, and define one or more alarms (low value, high value, etc.) with different messages, priorities, or contacts to notify.

Notify has been designed and developed to obtain maximum modularity, in such a way that it is easy to expand to new data acquisition systems and protocols. Likewise, the architecture of the application is multi-threaded, so that there are no locks or waiting in the operation of the different modules.

You will see along the manual that another main goal is to have a wide range of functions. In data capture you can apply basic or complex transformation formulas, masks, or binary offsets. In the alarm control it has multiple control conditions, stability times, customized messages, dependency control, and so on. When managing contacts, you will note that the configuration options are very powerful. The simple and usual cases are easy to setup, but the application allows you to go as far as the user needs.

Furthermore, another design goal has been that Notify should be a tool that the end user himself can install and configure, without requiring great technical knowledge or the intermediation of high qualified personnel.

Finally, the licensing system aims to be flexible enough, so that the user only acquires the functions that he really needs. We put at your disposal a **DEMO mode** of the application (trial version), fully functional for a certain time, which can be useful to previously evaluate if this tool is adapted to your needs. At Tecmatia we are convinced that this will be the case.

¹ OPC-DA: Ole for Process Communication, Data Access. Also known as Classic OPC, based on Microsoft's COM / DCOM technology.

² OPC-UA: OPC Unified Architecture. New OPC standard, integrating the functionality of previous protocols, but dispensing with COM / DCOM, in favor of open and independent protocols so that it works on other non-Microsoft systems.

³ Modbus: Communications protocol based on the master / slave (RTU) or client / server (TCP / IP) architecture, designed in 1979 by Modicon for its range of programmable logic controllers (PLCs). Converted into a de facto standard communications protocol in the industry, it is the one with the highest availability for the connection of industrial electronic devices.

2 Installation

2.1 System requirements

In order to install and run Notify properly, the requirements that the computer system must meet are, at a minimum:

- ✓ Windows® XP or higher operating system (Windows Vista or higher recommended).
- ✓ Pentium® type CPU or higher (Intel i3 or higher recommended).
- ✓ 1 GB of RAM.
- ✓ 100 MB of free hard disk space.
- ✓ Internet connection to send email or SMS through online gateways.
- ✓ GSM modem compatible with standard AT commands over serial COM port, for sending and receiving SMS.

2.2 Installing the application

Notify comes with an executable installer, which makes easy the installation process. It will guide you through the following steps:

- Select the installation language.
- Read and accept the license agreement.
- Select the destination folder. By default, a folder is suggested within "Program Files". It is recommended to use this folder.
- Specify the name of the Start Menu folder where to place a shortcut.
- Check or uncheck the option "*Allow to collect anonymous usage statistics*". If an Internet connection is available, Notify may collect certain application usage data to help Tecmatia understand the use and improve the product. Data that is collected are, for example, the total number and type of data clients and variables, types of messaging enabled, Windows version, PC characteristics such as processor and installed RAM, language, or screen resolution.

The installer gives you the option to launch Notify upon completion.

2.3 License

The use Notify you need a valid license for it. If this license is not activated, or it is not valid because it does not match the hardware or exceed the number of allowed elements, the application automatically activates the "DEMO" mode. In this mode, data logging and alarm notification are disabled after 30 minutes, and the application closes at 2 hours.



Some of the features described in this manual are only available for certain editions of Notify, or under license. Consult Tecmatia or your distributor if you need to enable more features.

2.3.1 Request a license

In *DEMO* or trial mode, the *Free* edition can be activated for free without the need to request a license or register the application. This version allows the declaration of up to 5 variables / alarms with

email notifications only. If this version is enough for you, please find in the next section how to proceed to activate this license.

If you want a license for a greater number of variables or to enable other additional features, please contact Tecmatia or an authorized distributor. Please note that each license is valid for a unique and specific computer. There are two methods of license provisioning and activation:

- **Online:** You will need an Internet connection from the PC on which the license is to be activated. In this case, you will be supplied with an activation key (20 alphanumeric characters).
- **Offline:** If you do not have an Internet connection from your PC, you will need to supply an identifier (hardware ID) of the computer where the software is being installed. You can obtain this hardware ID clicking the button "License → *Request license*" from the "About..." window of Notify.



Each license is valid for a unique and specific computer.

2.3.2 Activate the license

To activate a license, access the "About..." window, from the left side menu of the application.

- **License: Free:** You can activate this license only if you do not have any. It does not require any external license file, simply click on the button "*License → Activate FREE*" in the "About..." window and restart the application for the changes to take effect. You have the Restart option on the left side menu.
- **Other licenses (Lite, Standard, Professional, Enterprise...):** These licenses require activation, either through an activation key (requires an Internet connection), or through an external file supplied by the application provider. To activate the license, click on the button "*License → Activate license*" from the "About..." window and follow the steps indicated. For the changes to take effect it will be necessary to restart the application.

2.4 Update the app

If you have an Internet connection, from inside the application you can check if there are new versions available. You will find this function in the window "About ..." -> "Update ...". It will inform you if there are later versions and offers you the possibility to download and run the corresponding installer.

The version update is free during the first year of license, or for a longer period if you have purchased the maintenance option. It is advisable to always use the latest version available, as it may correct errors or provide new features that may be of interest to you.

If you do not have an Internet connection on the same PC, you can go to the *Downloads* section of the Tecmatia website from another computer and download the latest installer available. Copy it and run it on the computer where Notify is installed. Before proceeding with the update, it is advisable to close the application manually.

When updating the version, Notify automatically creates a backup copy of the current project settings, which can be automatically restored in case it is necessary to return to the previous version for any reason.

After the update period available in your license, if this is limited, the application will not allow automatic updating. If you manually install a more recent version, the license will be detected as "invalid", making it necessary to purchase a new one or reactivate the existing one, or return to the previous version of the application.



In the "About ..." window you will find the "Update ..." option that allows you to check if the application version is the latest available. It is recommended to check it periodically, or if you detect any anomalous operation or bugs that could be fixed in new versions.

3 Starting the application

Once the application is installed, you can start it using the shortcut in the Windows start menu. By default, the application starts minimized in the Windows system tray (notification area). To access the application, click on its icon, and the main screen will appear, from which you can access all the functions. Instead, if you right-click on the icon in the system tray, you will see a context menu with access to the main functions. Finally, if you hover over the icon, you will see a hint with the status.

The icon in the system tray will flash if there are active alarms. If there is any kind of error in the operation of the application, you will see a warning sign over the same icon.

Since Notify is designed to monitor alarms continuously, it will be common for the application to start with Windows. You can enable this behaviour from the main Settings window, under the Windows Integration options. In certain cases, it may be useful for the application to start with a certain delay after system boot, for example to allow a SCADA software to start⁴, or the server has valid data. In the same configuration window, you will find a parameter that allows you to adjust this time.



It is recommended to edit the settings of the Windows notification area so that the Notify icon is always visible, so you can see the status of the application more quickly. In Windows 10, you can find these settings on the Settings screen → System → Notifications and actions → Select the icons that will appear on the taskbar.

It is also recommended to enable automatic startup with Windows, with a certain delay if necessary, from the main Settings window.



Illustration 1: Start window and system tray icon.

⁴ SCADA: Acronym for Supervisory Control And Data Acquisition (Supervision, Control and Data Acquisition). It is specialized software that allows remote control and supervision of industrial processes.

4 Interface

Notify has been designed to provide a simple and intuitive interface to the user, in addition to adapting to current Windows application design criteria.

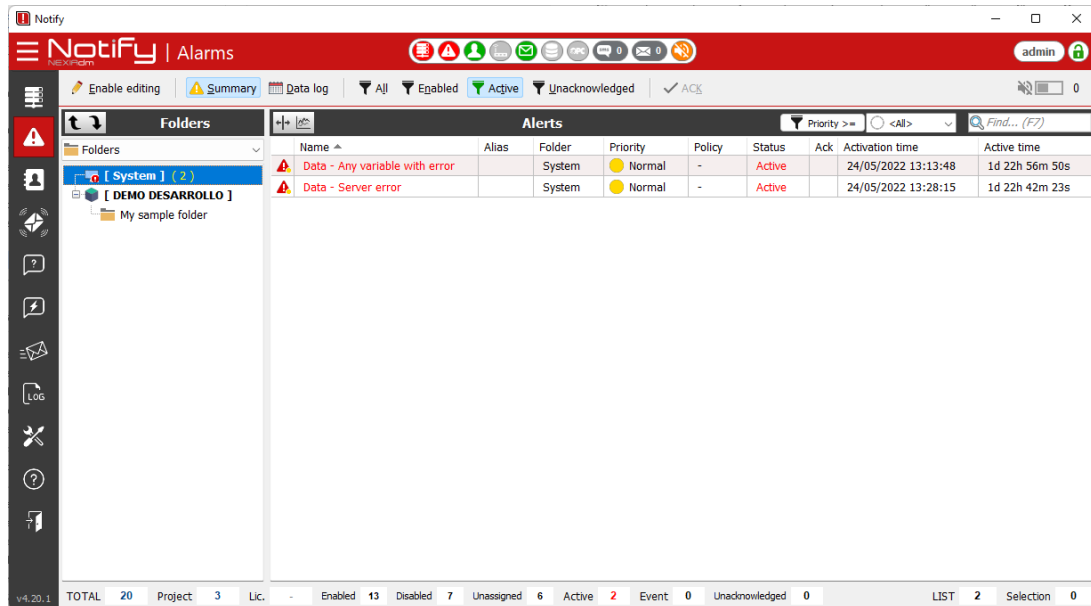


Illustration 2: Notify main screen

The side menu gives access to the main sections and modules. It can be expanded or collapsed to display or hide the name of each section.

At the top is the status bar, with icons that represent the status of the different modules of the application (as available in the license), according to the color code:

- **Red:** There is some error in the module, or active alerts.
- **Orange:** There is some warning, but it is not an error.
- **Green:** Everything works correctly. No errors or incidents.
- **Gray:** Service stopped or disabled.

Further, the number of SMS and emails in queue, pending to be sent, is shown in the status bar.

The icons on the status bar also allow direct access to the function, by clicking on them. For example, you can quickly access active alarms, or the operation panel of the different message delivery systems.

5 Security: User Control

Notify has three user levels:

- **Administrator:** Full access to settings and data. The username is preset as "admin" and cannot be modified. The default password is "admin", but can and should be modified from the main settings.
- **Operator:** Can view and consult data, but it is restricted in terms of configuration changes. The default username and password are "operator", but both the name and the password can be modified.
- **Guess:** When there is no authenticated user. You can barely view the most basic information, current values of variables and alarm status, but you cannot modify any settings or access historical data or contact information such as phone numbers, email addresses, etc.

Notify allows to link the user identification to an external system, which publishes the username and current level through an OPC-DA server. This is useful to link with the user management of a SCADA software. For example, Wonderware Indusoft Web Studio⁵ enable this feature. Even if you link to an external system, the "local" user ID will still work in Notify. However, this local user cannot be ported to the external application.

Furthermore, Notify can share users declared in the Grafana application, since it usually coexists with Notify in projects where data is logged in an external InfluxDB database.



After a configurable time of inactivity, Notify automatically closes the user session.

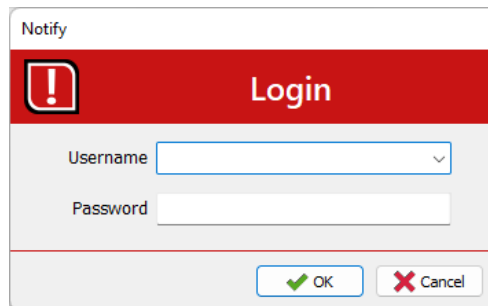


Illustration 3: User login

⁵ Indusoft and Wonderware are registered trademarks of Schneider Electric Software, LLC.

6 Variables

The main basis of the application is the acquisition of data on which the alarms will be managed. Notify currently supports these protocols and data services:

- API
- Database
- Calculated data
- Internal data
- Ethernet/IP
- Historical in DB
- Microcom Zeus
- Modbus (RTU, ASCII, TCP)
- OPC-DA (COM, XML)
- OPC-UA
- Network
- Siemens S7
- File system
- Operating system (Windows)
- Telegraf



Ilustración 4: Available data clients

Access the "Variables" section to query or edit data clients, groups, and variables.

Name	Folder	Status	Time	Value	Units	Alarms
Bucket Brigade.Int1		good	12:14:10	108		0 / 2
Bucket Brigade.Int2		good	12:14:10	108		0 / 1
Bucket Brigade.Real4		good	12:09:25	0,00		0 / 0
escribe aqui		disabled	12:09:23	Null		0 / 0
Prueba curva		good	12:14:10	9,80		0 / 0
Random.Boolean		disabled	12:09:23	Null		0 / 0
Random.Int1		good	12:14:10	108,0		0 / 0
Random.Int2		disabled	12:09:23	Null		0 / 0
Random.Int4		disabled	12:09:23	Null		0 / 0

Illustration 5: Variables section.

6.1 Query and visualization

From the *Variables* section, in normal mode, you can view the list of variables (right panel), with their settings, the current value, date of the last value change, reading status, etc. On the variables panel, there are two display modes:

- **Simple view:** Shows only the basic data (Name, Status, Current value). Columns are automatically resized to fit the available space.
- **Extended or custom view:** Shows all possible data. Allows you to select the columns you want to view by right clicking on the column title bar. In this mode, the width adjustment is not automatic, but you can force it using the *AutoFit Columns* button.

On the left is a navigation tree with several possible views according to Notify organizes the declared variables:

- **Data Clients:** Shows the list of declared servers and reading groups. From this view you can add and configure new servers, read groups, and variables.
- **Folders:** Shows a tree of freely configurable folders declared by the user. From this view you can organize the variables according to the criteria that best suits your installation (production processes, physical location, etc.). It will be the most common view.
- **Variable kind:** Variables are organized according to their type: Analog, Digital, Counter, etc.
- **Data type:** Variables are organized according to source data type: Boolean, Integer, Float, ...

The following views are focused on edition using *drag & drop*:

- **Enable:** Variables are organized into two groups: enabled or disabled. Useful in edit mode.

- **Writeable**
- **Units**
- **Decimals**
- **Data log...:** Groups variables depending on the configuration of the internal and external data log.

When selecting an element in the tree, the associated variables will be displayed in the main panel. Use the "Select All" option from the context menu to display the complete list of declared variables.

The Alarms panel shows the alarms configured on the selected variable.

The footer panel, in the Data Clients view, has relevant information about the selected server (name, start time, last data update, status, ...)

Finally, in the status bar you will see a series of counters with information on the total of variables, selected, enabled, or with reading error, as well as the maximum limit provided by your license.



The number of variables will be the main metric when selecting the license you will need.

6.2 Data client configuration

To configure data clients and variables, you must be logged in as an administrator user. In the *Variables* window, click on the "Enable editing" button, you will see that the display mode changes to *Extended View* and all the editing functions appear (see *Illustration 5*).



Note that Notify handles all the application process at the same time (data acquisition, alert control, registration, notifications, etc.) while allowing you to edit or modify any configuration.

By clicking on "Enable editing", in this and other windows, the application internally manages a copy of the elements that are still being used in "run time" with their previous settings. Changes are not applied to actual items until "Save" is clicked. If you press "Cancel", no changes will be applied and no interference will occur.

Before you can add variables, you need to add the data client, as well as create at least one reading group.

When adding or configuring a data client, in addition to the specific data of each protocol, there is some configuration common to all:

- **Name:** Text that identifies the data client.
- **Icon:** You can select an image that best identifies the server. By default, there is an image associated with each protocol.
- **ON / OFF:** It enables or not the data acquisition of this server. Cascading disables all the variables declared for this server, in such a way that their value will not be controlled, and therefore alarms will not be controlled. In the historian you will see a *Null* value.
- **Time for data stabilization after connecting:** On certain servers, the initial data after connection is not up-to-date or invalid (for example, in some cases all values start at "0").

Use this parameter to indicate a timeout during which the value will be ignored, preventing false alarms from occurring.

- **Time for data in "bad quality" if connection is lost:** If the connection to the server or the data source is lost, the data goes to bad quality for a while. If in that time the connection is not recovered, the values go to *Null*, and the state to disconnected.
- **Active connection control: Force new connection...**
 - **If the data does not change during...:** Check this option if you want to force reconnection if no value, or the value of a variable does not change for a certain time. Useful for example with a special variable that changes continuously to control a possible block in the data reading.
 - **If not responding for...:** Some servers allow you to control the state to know if it is responding or not (for example, OPC-DA, if it implements the functions required by the standard). Check this option to monitor the status, and force reconnection if it does not respond after a while.
 - **If no variable OK:** Forces reconnection if no variable can be read with *good* value, in the entire data client or in some group.

6.2.1 Data source types

The next step will be to select the protocol or type of server, for which there will be custom configuration fields:

6.2.1.1 API

Allows reading data from external APIs. Currently has support for:

- **AEMET:** Agencia Estatal de Meteorología (Spain) [<http://www.aemet.es>]
- **IPMA:** Instituto Português do Mar e da Atmosfera (Portugal) [<https://www.ipma.pt/>]
- **MeteoGalicia** (Spain) [<https://www.meteogalicia.gal>]
- **OpenWeather** [<https://openweathermap.org>]
- **REE – ESIOS** (Red Eléctrica de España) [<https://www.esios.ree.es/>]
- **Sunrise Sunset** [<https://sunrise-sunset.org/>]
- **Meteostat** [<https://meteostat.net/>]
- **Servei Meteorologic de Catalunya – Meteocat** [<https://www.meteo.cat>]

The configuration required is specific to each API. In some cases it requires obtaining an API key by registering on the web.

Illustration 6: External API client settings.

6.2.1.2 Database

With this client you can access information in traditional databases, also known as SQL or RDBMS. It natively supports Oracle, SQL Server, PostgreSQL, MySQL/MariaDB, Firebird/Interbase, and SQLite. And indirectly any ODBC connection.

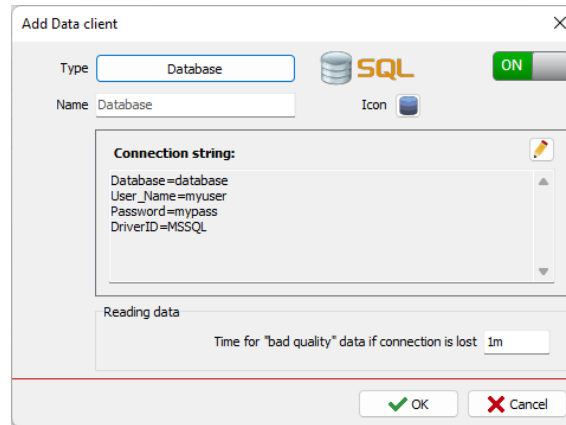


Illustration 7: Database client settings.

In editing mode, use the button with the pencil icon, or the F2 key, to open the database connection parameter editor. First of all, be sure to select the correct driver as the rest of the parameters depend on it.

- **Ora:** Oracle Database.
- **MSSQL:** Microsoft SQL Server.
- **PG:** PostgreSQL.
- **MySQL:** MySQL, MariaDB.
- **FB:** Firebird, Interbase.
- **OBDC:** Other databases that have ODBC access.

Fill in the data required by the selected driver (server or host, port, database, username, password...). Use the "Help" button or your database server documentation for help with the necessary settings. If necessary, use the "Revert To Defaults" button to return to the default configuration parameters.

6.2.1.3 Calculated data

It allows obtaining calculated or aggregated data from other variables or alarms. For example, a variable on this server can be the sum, maximum, mean, or other operations on a certain group of variables. Another example, a counter of activations of one or more alarms or digital variables. At this level, no configuration other than the name is needed.

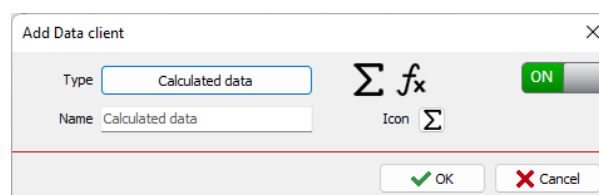


Illustration 8: Calculated data client settings

6.2.1.4 Internal data

It allows obtaining internal data from the Notify application itself, such as the count of variables or alarms, status or global information, or by services, etc. The instance can be local or remote, connecting to another Notify instance via REST API. In this case, you must indicate the server address (name or IP), the port, as well as an API key with read permission.

The screenshot shows the 'Add Data client' dialog box. The 'Type' field is set to 'Internal data'. The 'Name' field is 'Internal data'. There is a 'Notify' logo and a green 'ON' button. Below the name field, there are radio buttons for 'Local' (selected) and 'Remote'. At the bottom, there are 'OK' and 'Cancel' buttons.

Illustration 9: Internal Data client settings.

6.2.1.5 Ethernet/IP

Protocol typically used in PLCs from Allen-Bradley / Rockwell, Omron, and other manufacturers.

Configure the following fields:

The screenshot shows the 'Add Data client' dialog box. The 'Type' field is set to 'Ethernet/IP'. The 'Name' field is 'Ethernet/IP'. There is an 'EtherNet/IP' logo and a green 'ON' button. Below the name field, there is a dropdown menu for 'PLC'. Below that, there are three input fields: 'Host / IP', 'Port' (set to 44818), and 'Path' (set to 1,0). At the bottom, there are 'OK' and 'Cancel' buttons.

Illustration 10: Ethernet/IP client settings.

- **PLC:** Choose the family of the specific PLC controller, among those supported. If it does not appear in the list, it is possible that one of the existing families works.
- **Host/IP:** Enter the IP address or hostname of the controller.
- **Port:** Specify the TCP/IP port number. Default 44818.
- **Path:** See the specific PLC documentation for information on the use of this field. The default value is "1.0".

6.2.1.6 Historical in DB

This data source allows you to obtain statistical data from a list of samples stored in a database.

These database types are supported:

- **Internal data (Notify):** The data is obtained from Notify's own internal history, according to the availability and limits of the license. You can select variables or alarms, if they are configured for internal logging.

- **RDBMS / SQL:** Allows reading from relational or traditional databases (Oracle, SQL Server, PostgreSQL, Firebird, MySQL, ODBC, etc.) Configure the connection string, selecting the appropriate driver and indicating the needed connection parameters.
- **InfluxDB:** For data stored in this database of type TSDB. You will need to configure the connection settings (server, port, database, username, password).

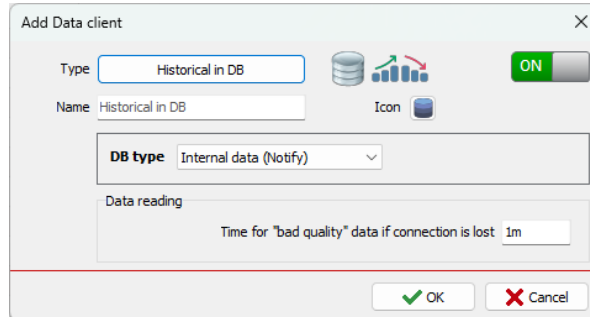


Illustration 11: Historical in DB client settings.

6.2.1.7 Microcom Zeus

It allows collecting data from Microcom datalogger equipment, through its Zeus platform. It can be from a local instance, directly from the database (typically SQL Server), or through its public API.

Configure the following fields:

- **Connection type:** Choose whether to read the data directly from the database (local instance) or through the API (remote instance).
- **Connection settings:** Use the button with the pencil icon, or the F2 key, to open the editor for database- or API-specific connection parameters. For API access you will need an authorized user account.

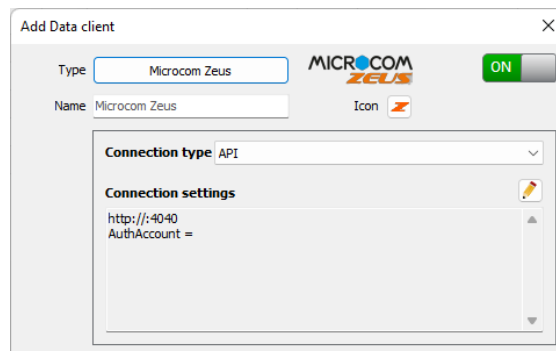


Illustration 12: Microcom Zeus client settings.

6.2.1.8 Modbus

Modbus can use serial communication or TCP/IP as the physical means of transmission, also referred to as Modbus RTU or Modbus TCP, respectively. Notify supports all Modbus variants: *RTU*, *ASCII*, *TCP*, or *RTU over TCP*.

Configure the following fields:

- **COM, Bits per second...** : For RTU and ASCII modes. Configure the serial port to be used, as well as the basic parameters of speed, data bits, stop bits, and parity.

- **Host/IP, Port:** For TCP modes, configure the IP address or hostname as well as the listening port (default for Modbus is 502).

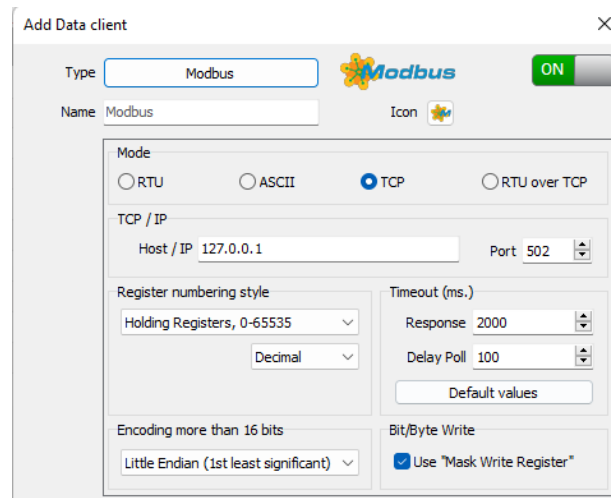


Illustration13: Modbus client settings.

- **Register numbering style:** Although the address used in the Modbus frame is standardized from 0 to 65535, and is the same in any system, at the level of documentation and visualization the modbus address space and nomenclature is not entirely standardized. There are different numbering schemes, some of them dragged from the Modicon protocol, predecessor of the current Modbus. Refer to your device's documentation to select the one that works best:
 - **40001-49999 (Modicon 5 dig.)** : Where 40001 corresponds to register 0 of Holding Registers at the Modbus protocol level, and only 9999 registers of the available 65535 are addressed.
 - **400001-465536 (Modicon 6 dig.)**: Same as above, but with 6 digits to address all possible records.
 - **4:00000-4:65535**: The data block is separated from the address by ':', starting at 0.
 - **4:00001-4:65536**: Same as the previous one, but starting to number in 1, which corresponds to the direction 0 in the Modbus plot.
 - **Holding Registers, 0-65535**: The data block is displayed in text.
 - **Holding Registers, 1-65536**
 - **Decimal | Hexadecimal**: How to display the register number.
- **Encoding more than 16 bits:** In Modbus the base data type is the word, that is, values of 16 bits (2 bytes), which in integer value supposes values in the range 0-65535. Depending on the device or manufacturer, it is possible to represent data types of more bits, using several words. In this case, it is necessary to indicate to Notify what is the order in which the data represents, depending on greater or lesser weight first, a concept that is known by its name in English "*endianness*", and that can be *Big Endian* or *Little Endian*, if the word of greater or lesser weight is first encoded, respectively.
- **Timeout: Response:** Maximum waiting time for the response to the Modbus request.

- **Timeout: Delay Poll:** Time between consecutive requests in case of multiple requests in the same data group.



Use the "Defaults" for quick and regular configuration.

Modify the times only if you experience problems with the Modbus TCP or RTU connection.

- **Bit/Byte write – Use "Mask Write Register":** For writing data types "bit" or "byte" within a word of 16 bits can be used two methods. On the one hand, it is possible to modify the bits of the word locally, on the client side (Notify), and send the result by typing the entire new word. But this can cause a problem if other bits that may have been modified on the server side during the process are overwritten. To avoid this, the Modbus standard has the "Mask Write Register" function, which modifies only the necessary bits, and keeps the rest unaltered. It is recommended to use whenever possible, and only uncheck in case the slave device does not implement this function.

6.2.1.9 OPC-DA

Probably the most widely used data communication protocol in industrial automation and SCADA applications.

The screenshot shows the 'Add Data client' dialog box with the following settings:

- Type: OPC-DA
- Name: OPC-DA
- Protocol: COM
- Computer Name: [empty]
- Server Name: [empty]
- Class ID: [empty]
- Arrays: tag_name[n]
- Time for data stabilization after connecting: 0s
- Time for "bad quality" data if connection is lost: 1m
- Active connection control: Force new connection ...
 - If the data does not change during ...
 - This variable
 - Any variable
 - If not responding for ...
 - If no variable OK ... in the entire data client.

Illustration 14: OPC-DA client settings

Notify supports data acquisition through 3 variants of the OPC protocol:

- **COM:** Also known as classic OPC, or OPC-DA. It is based on Microsoft's COM / DCOM technology. It is possible to connect OPC servers over the network, however, the configuration at the operating system level is complex for the end user and will not be covered in this manual. In the case of connection through the network, it is preferable to use the XML protocol or the UA if they are available.
- **XML:** Replaces COM / DCOM technologies used by classic OPC by HTTP / SOAP and Web Service technologies. However, its use has not spread and today it has been superseded by the more modern OPC-UA variant.

- **UA:** Also abandons COM / DCOM in favour of two transports: SOAP / HTTP(S), allowing its implementation in any operating system. It is the most modern version of OPC.

Configure the following fields:

- **Protocol:** Select the COM or XML protocol as needed.
- **Server Name:** Use the "... " to access the pop-up window where you can configure access to the OPC server in detail. Typically, the OPC server will be selected from the list of available servers, as an important feature of OPC is the ability to publish access data. If the server is not in the list, you can enter the data manually as the fields are editable.
- **Arrays:** Choose the *array* numbering format that best suits you. In this way Notify will handle the variables declared with this syntax as *arrays*, or as *individual items* or *tags* in another case. You can also indicate the preference of the range of indices, from 0 to N-1, or from 1 to N, with N being the total of elements of the array, so that later it is easier for you to configure variables. The default setting is "*tag[n]*", with *n* being in the range 0..N-1.

6.2.1.10 OPC-UA

It is the most modern version of OPC and with the greatest future projection. It also abandons COM/DCOM in favor of two transports: SOAP /HTTP(S), allowing its implementation in any operating system.

Configuration parameters are common to OPC-DA, so you can go to the previous section for a description of these.

Note that in OPC-UA there is the possibility of two-dimensional arrays, in which case the expected syntax is "*tag[n,m]*".

In the case of OPC-UA, probably you will need to configure some security aspects such as authentication, encryption, etc., that your server requires. For authentication with certificate, it is possible to use the internal Notify certificate (self-signed, and unique for each instance), or to supply your own.

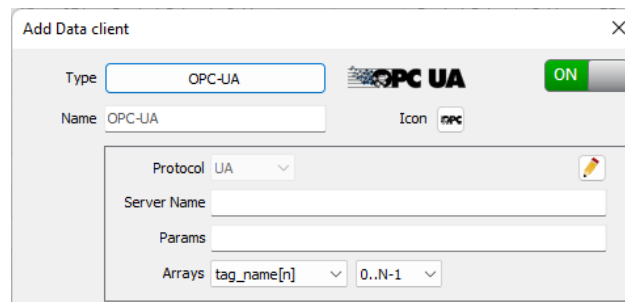


Illustration 15: OPC-UA client settings.

6.2.1.11 Network

This data client does not require additional configuration at this level. It allows to declare variables that control certain network data (Ping, TCP, FTP, ...)

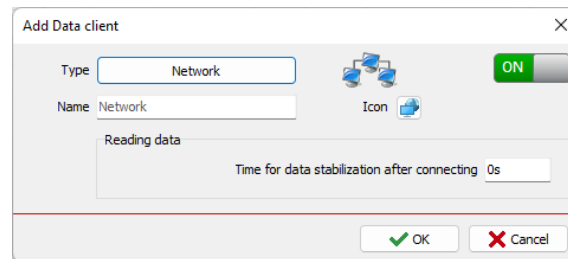


Illustration 16: Network data client settings.

6.2.1.12 Siemens S7

Notify supports the acquisition of data from Siemens SIMATIC S7 controllers via Ethernet.

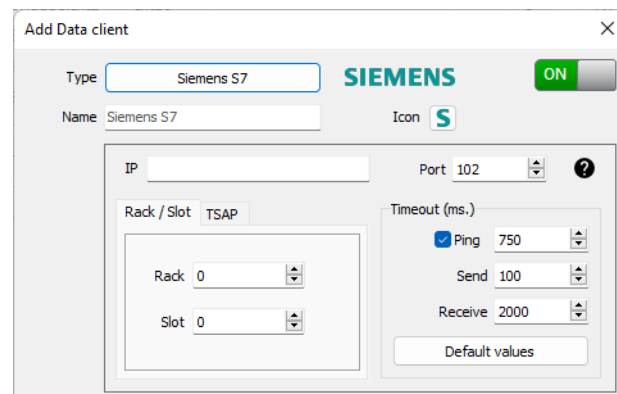


Illustration 17: Siemens S7 client settings.

Configure the following fields:

- **IP:** Enter the IP address of the computer where the controller resides.
- **Port:** Enter the TCP / IP port number where the controller resides.
- **Rack / Slot:** For S7 300/400/1200/1500/WinAC CPUs add the rack and slot number of the controller. The typical settings are:
 - S7-300: Rack 0 - Slot 2
 - S7-400: It depends on the hardware settings (most common is Rack 0 - Slot 3).
 - S7-1200: Rack 0 - Slot 0
 - S7-1500: Rack 0 - Slot 0
 - WinAC: Rack 0 - Slot depends on the controller settings.
- **TSAP:** For S7 200 CPUs and LOGO, write the Local and Remote TSAP of the controller in hexadecimal.

- **Timeout - Ping:** By checking the “Ping” option, Notify will make a more intelligent and efficient management of the connection to the PLC, establishing the S7 data connection only if the PLC responds to a Ping. It is recommended, but not required.
- **Timeout - Send:** Maximum time for sending an Ethernet data packet.
- **Timeout - Receive:** Maximum time for data reception.

6.2.1.13 File system

This data client also does not require additional configuration at this level. It allows to declare variables that control certain data of files and folders (exists, size, dates ...)

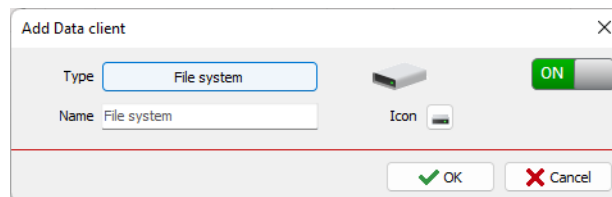


Illustration 18: File system data client settings.

6.2.1.14 Operating system

It allows acquiring system data (processes and services, use of resources, user activity, etc.).

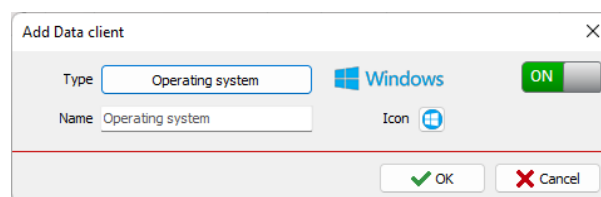


Illustration 19: Operating system data client settings.

6.2.1.15 Telegraf (InfluxData)

This client allows data acquisition through InfluxData's Telegraf agent, an open source software that has a multitude of input plugins for very different data sources (databases, operating system, cloud platforms, meteorological services, IoT, etc.). This provides a high degree of flexibility to Notify, as any data that Telegraf may acquire will be readily available. You can download Telegraf directly from the InfluxData website.

<https://www.influxdata.com/time-series-platform/telegraf/>

The configuration is simple since the software consists only of an executable and a configuration file. Notify opens a listening socket and Telegraf must send the data to it using the *Socket Writer* output plugin. You can find detailed help on the Telegraf website.



Among the plugins available for Telegraf you will find some of special utility, such as weather data from OpenWeatherMap, system data (CPU, disk, memory, network ...), IoT, MQTT, SNMP, etc.

To configure Notify, you will find this screen:

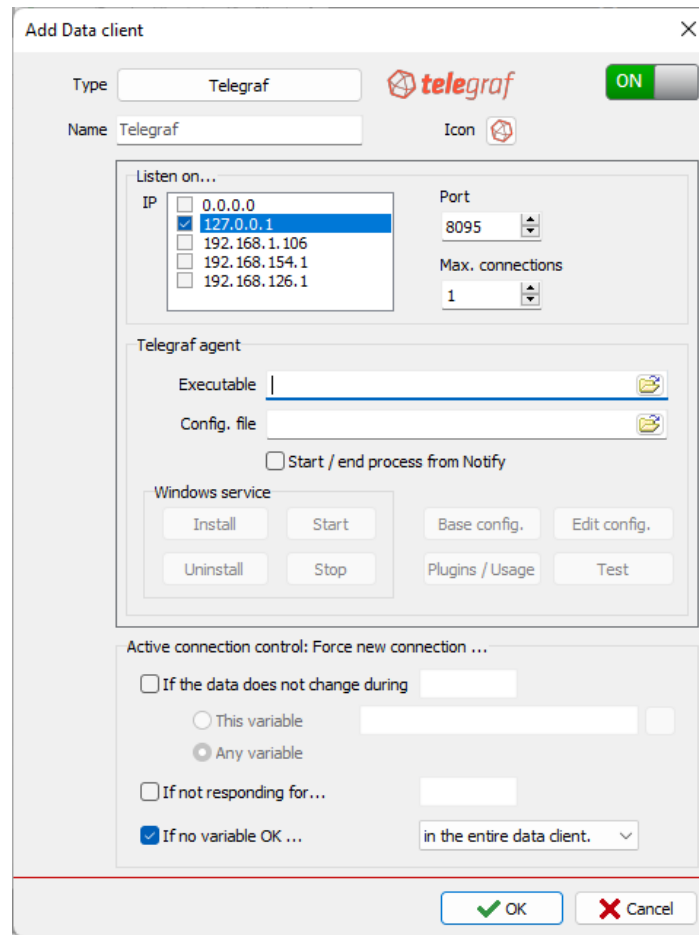


Illustration 20: Telegraf client settings

- **Listen on...:** Select the IP or network interfaces, and the port number, to open the listening socket. It is also possible to limit the maximum number of active connections.
- **Telegraf agent:** Although they are not required, if the local paths to the executable and configuration file are indicated, some utilities are available from Notify to help install and configure Telegraf, either as a Windows service or as an independent executable process.
- **Start / end process from Notify:** If Telegraf is not installed as a Windows service, starting and stopping the executable process can be delegated when connecting / disconnecting the client in Notify.
- **Windows service:** If you prefer to install and run as a service, here you have the quick access to install, uninstall, start, or stop the service manually.
- **Base config.:** Generates a basic Telegraf configuration file with the selected input plugins.
- **Edit config.:** Shortcut that allows quick edition of the configuration file.
- **Plugin / Usage:** Selecting an input plugin, it shows help about its configuration and use.
- **Test:** Allows you to test the current configuration, showing the result.

6.2.2 Reading groups

In Notify, a set of variables must be associated with a reading group, which gives common properties to the set. The main objective of the reading groups is to differentiate the refresh times of different variables based on the priority or the speed of change in the real data. However, you can

create several groups with the same refresh times if you are interested in separating the variables by zones or any other criteria that you consider useful or intuitive.

Illustration 21: Reading group settings.

When adding or configuring a reading group, a window is displayed in which you can configure:

- **Name** of the group of variables.
- **Data refresh interval**: How often the data will be read.
- **Refresh at o'clock**: It forces the reading to be done in synchronized moments. For example, if the interval is 1m, the reading will be made on the second ":00" of each minute. If it is every 30s, in the second ":00" and in the ":30". If it is not checked, as it works by default, the reading will be done each time interval from the moment of start of Notify or activation of the client, independent of the seconds.
- **ON / OFF**: Allows disabling the data refresh of the entire group. If you disable a group or variable, its value will not be controlled and therefore alarms will not be controlled, and you will see a *Null* value in the historian.



To optimize the performance of the system, it is recommended to separate the variables into different groups with refresh times appropriate to the type of data, its rate of change or its relevance. If your installation has a large number of variables, avoid using millisecond refresh times in a generalized way.

For many types of data sources, no additional configuration is required for the reading group. However, for some that are detailed below, it is possible or even required:

6.2.2.1 API

Configure the required fields according to the configured API. For example, for a weather API:

- **Station**: Choose the weather station from which you want to collect the data.
- **Location**: Optionally, the location of the station will be obtained automatically if available, or you can indicate it manually, in longitude and latitude.

6.2.2.2 Database

- **SQL**: Specify the SQL query from which to obtain the values of the group variables. Click on the pencil icon to open the advanced editor. You can indicate an SQL as is or use *parameters* (when executing they are sent to the database engine, which replaces them with values, normally used in the clause *where*) or *macros* (before executing they are replaced by texts, normally used for field or table names).

Available parameters:

- `:serverName`: Name of the data client.
- `:groupName`: Name of the reading group.

- *:groupNameNumber*: Number contained in the name of the reading group.
- *:vXXX*: Value of a variable, indicating the internal ID (for example, :v123)
- *:aXXX*: Status of an alarm, indicating the internal ID (for example: a123)

Available macros:

- *&serverName*: Name of the data client.
 - *&groupName*: Name of the reading group.
 - *&fieldList*: List of all field names referenced in all the group variables, separated by commas.
 - *&indexFieldName*: Name of the index field, if configured.
 - *&indexValueList*: List of all values used in the index fields of the variables contained in the group.
 - *&indexValueListQuoted*: Same as above, but values with simple quotation marks, useful for text type field.
- **Index field**: Optional. It allows you to indicate the name of a field, returned by a multi-record query, which will be used to filter the result to assign a specific record to the corresponding variable, in which you will configure a specific value for this index field .
 - **Warning if the query takes longer than ...**: Optional, used for additional visual control only. Flag warning on reading group if query execution takes too long.

6.2.2.3 Historical in DB

At the group level, in addition to the refresh interval, the start and end dates of the data to be consulted, common to all the variables of the group, are configured.

- **Date range (Start/End)**: Configure the time range of data to be read, which will be dynamically calculated relative to the current date.
- **Adjust start/end**: If this box is not checked, the start and end date are calculated by subtracting a certain number of minutes, hours, days,...from the current date. If "Adjust..." is checked, the start and/or end date will be rounded to the beginning or end of the interval. For example, with "Last 1 hour", and it is now 19:15, without checking the box the query will be from 18:15. If adjusted, it will be from 18:00.
- **Hour range**: Optionally, certain hours can be filtered within the date range. In this way, only statistical data of the samples is obtained from that time window, ignoring the others.

6.2.2.4 Microcom Zeus

- **Station**: Select the required station from the drop-down or type the name.

6.2.2.5 Modbus

- **Slave address**: Indicate the slave address to read. For RTU and ASCII modes.
- **Unit ID / Ignore Unit ID**: Equivalent to the slave address for the Modbus TCP variant. If it is not necessary, check "Ignore Unit ID".
- **Offset in register address**: The register number of each variable is moved by adding the indicated offset. Useful for setting up multiple repetitive groups.

- **Optimize reading in blocks:** Instead of making a Modbus request for each declared variable, in order to optimize the times and the reading process, Notify groups the registers to be read with neighbouring addresses in blocks. Although this grouping works correctly with the default values, it is possible to customize it to specific needs:
 - **Max. registers per block:** It is the maximum number of registers (16-bit words) that will be read in a single request.
 - **Max. gap between registers:** It is the maximum gap between non-consecutive records, but close in direction, when grouping the readings. In other words, it corresponds to addresses where there are no variables declared in Notify, but it is optimal to include in the request to minimize the number of reads required to obtain the total number of variables.
- **Override dataclient settings:** In certain cases, for example when devices of different types are connected to the same serial bus, it may be necessary to adjust the register numbering format, or the encoding of more than 16 bits, etc., independently for each device.

6.2.2.6 File system

- **Base path:** Optional. Specifies the root directory that will be used to obtain the status of certain types of variables within the reading group.

6.2.2.7 Telegraf (InfluxData)

- **Measurement:** Indicate the measurement name from which to extract data for the variables contained in the group. If the Telegraf agent is already configured and operational, it allows pre-selection from the list of measurements received. If not, it is necessary to know the name previously, either using *Test* button, or by consulting the specific plugin documentation.

6.2.3 Editing

- **Add folder (from tree):** Besides the reading groups (required), you can optionally organize the variables in a tree of freely configurable folders, to better adapt to the organization criteria you consider appropriate (by process, by location, etc.). To do this, select the *Folders* view. Here you can freely add, remove, rename, or move these groups. Variables that do not have a folder declared appear under the *<Unassigned>* node.
- **Edit:** Allows you to modify an existing server or reading group. The data that you can modify is the same as when you created it. It also has a quick shortcut to rename (F2).
- **Remove:** Deletes the selected tree elements (servers, groups or folders).
- **Undo:** Allows you to undo the last operation performed. If you need to undo multiple steps, you will have to cancel your changes and resume editing.
- **Enable / Disable data client**

Just as it is possible to disable a group, it is also possible to disable an entire server without removing it. This feature can be useful during initial setup, or to configure the application from a different computer than the final installation.

If you use this option from Edit mode, the server is permanently disabled. However, in Normal mode the change is temporary, and the server recovers its settings when you restart the application.

6.3 Variables

6.3.1 General

Before adding variables, you must first select the reading group to which they will belong, and then select "Add variables" in the upper button panel, or in the context menu.

Once you have registered a data as a variable in Notify, you can configure it to suit your needs or criteria. You can access the window by double-clicking on the variable in the list of variables, or by selecting the variable and then pressing *Edit*.

You can edit the following fields, common to any type of variable, independent of the protocol:

- **Server / Group:** Name of the server and reading group to which the variable belongs. It cannot be modified directly from here once the variable is created. If you need to move a variable to another read group, use the drag and drop function from the variable list.
- **ON / OFF:** Allows you to enable or disable the variable, instead of deleting it.
- **Name:** Text that will serve to identify the variable later throughout the system. You can also use it as text when sending alarms. Unique names are recommended, but not required.
- **Alias:** Optionally, you can indicate a short text that identifies the variable, which will be used in SMS messages or writings, queries, or alarms. If you do not specify an alias, the full name will be used. Aliases are always set to uppercase.
- **Mode (Read Only / Read-Write):** Indicate whether or not it will be allowed to change the value of the variable from Notify, either from the application interface, the OPC server, the REST API, or actions or messages sent to the application (SMS or Email).
- **Folder:** Enter the path of the folder where to place the variable in the Folders view tree, using the "/" separator for subfolders.

6.3.2 Display

- **Type:** Indicates how the read value will be treated and displayed:
 - *Digital* or binary (0/1)
 - *Analog:* Value that can vary within a certain range.
 - *Counter:* Increasing value that corresponds to a meter measurement (for example, m3 or kWh consumed).
 - *Number:* Numeric value, no units or graphics style.
 - *Hexadecimal:* The value is displayed in hexadecimal format.
 - *Text:* The value read will be processed and shown as text.
 - *Date:* Interprets the value as a date and is displayed in the specified format.

Note that this is only a display setting, independent of the source data type. For example, if a variable is configured as *Digital*, in Notify the value will always be 0 (*false*) or 1 (*true*) depending on the original data being zero or non-zero. Depending on the type of variable, other configuration parameters will be available or not, as well as specific query functions. For example, for *Digital* or *Counter* type, Statistics mode is available in graphs and reports.

Digital variable

- **Text 1/0:** It allows associating an alternative text for the value 1 and 0. You can use default values such as *On/Off*, *Active/Inactive*, *Ok/Error*, *True/False*, *Yes/No*, or enter any other custom text.

Analog variable or counter, you can configure these parameters:

- **Decimals, Units:** Used to format the displayed value.
- **Range (Minimum - Maximum):** Used as full scale in graphs, for example.
- **Graph style:** Select how you want to display the graphs of this variable, Staircase or Linear, as best suits the physical measurement to be measured. This is a default setting, you can always manually change the display in the graph component.

6.3.3 Acquisition > Transformation

For advanced configuration, it is possible to apply one or more transformation operations on the original value read to obtain the final value of the Notify variable. There are a lot of available operations, grouped into families:

- **Arithmetic:** Mathematical operators.
 - $+$, $-$, $*$, $/$, $\%$, *Sqrt* (square root), $^$ (exponent), *mod* (integer division remainder), *ln* (natural logarithm), *log10* (base 10 logarithm), *Absolute value*
- **Binary:** Bit-level operators.
 - *Bit:* Take the state of a bit. Indicate the number of bit to read, where 0 is the least significant bit.
 - *Not:* Negated value.
 - *Mask:* Reads certain bits ignoring the value of others. Enter the mask in hexadecimal format.
 - *And:* The result will be a digital variable (0/1), which will be 1 if all the bits indicated in the mask are 1, and 0 otherwise.
 - *Or:* The result will be a digital variable (0/1), which will be 1 if any of the bits indicated in the mask is 1, and 0 if all of them are 0.
 - *Shr >>:* Bit shift right.
 - *Shl <<:* Bit shift left.
- **Bytes:** Byte-level operators:
 - *Hi (byte):* Returns the highest byte of a Word or 16-bit data.
 - *Lo (byte):* Returns the lowest byte.
 - *Swap bytes:* It switches the position of bytes 0 and 1, if the data is 16 bits, and bytes 3 and 4 if the source data is 32 bits.
 - *Swap word:* For 32-bit data, it switches the position of 16 bit blocks (word).
 - *Swap BW:* Combine the two previous operations, switching bytes and words.
- **Round:** Rounding operators, limits, etc.
 - *Round:* Rounds a floating-point value to a specified number of decimal places, or to its nearest integer if the decimal places are set to zero.

- *Trunc*: Returns the integer part of a floating-point value.
- *Floor*: Round the value to the lower integer. For positive values it is equivalent to *Trunc*, but not for negative values: *Trunc* (-2.8) = -2; *Floor* (-2.8) = -3
- *Ceil*: Rounds the value to the highest integer. *Ceil* (-2.8) = -2; *Ceil* (2.8) = 3
- *Min, Max*: It gets the lower or higher value between the one read and the one indicated.
- **Filter**: Input data filtering operations.
 - *Minimum/maximum limit*: Sets a minimum or maximum value. If the value read exceeds this value, it stays in this.
 - *Ignore value / range / out of range*: Ignores a specific input value, or a range of values, keeping in that case the previous value.
- **Trigonometry**:
 - *Sin, Cos*: Sine or cosine of the value read. Available for values in radians or degrees.
- **Comparison**:
 - *Odd, Even*: Returns 1 or 0 depending on the value is odd or even. For floating-point values, only the integer part is considered.
 - =, <>, >, >=, <, <=: Compares the value with a given setpoint and returns a value of 1 (true) or 0 (false) if matches the condition or not.
 - *Range*: Returns 1 if the value is within a specified range. Enter the range in the format "m..n" or "m,n", where n is greater than m. The result will be 1 if "m <= value <= n", 0 otherwise. If you need an out of range condition, you can add a *Not* operation just after the *Range* operation.
 - *Values*: Returns 1 if the value of the variable is in the specified list of values. Indicate the values separated by commas.
 - *Null*: Returns 1 if the value of the variable is *Null*, that is, it has no value.
- **Date**: Operations with date type data.
 - *Second, Minute, Hour, Day, Month, Year, Sec of Day, Min of Day, Day of Week, Month Day, Year Day, Week*: Gets the corresponding part of the date, as a numeric value.
 - *+ seconds, + minutes, + hours, + days*: Increase the date by the indicated number of seconds, minutes, hours, or days, respectively. Decreases if the entered value is negative.
 - *Trunc minute, hour, day*: Truncate the date to the level indicated. In other words, it zeroes the data beyond that level. For example, "15:10:45" truncated to hour would be "15:00:00".
 - *From Unix Time, To Unix Time*: They allow the conversion of date in numeric format of Unix time (number of seconds elapsed since January 1, 1970). Indicate whether the read date refers to UTC or the current time zone.
 - *Dif. days now*: Returns the difference in days between the value's date and the current date. To get minutes, hours, etc., use an arithmetic operation just after.
 - *Format*: Converts the date to text, with the indicated format. It must be a valid format for Delphi's *FormatDateTime* function.

- **Text:** Operations for text values.
 - = **?*: Checks if the text matches an MS-DOS pattern, returning 1 (*true*) or 0 (*false*). The pattern must be entered with these wildcards:
 - ***: Represents anything, up to any number of characters and digits. Ex: *P** matches any text that begins with "P".
 - *?*: Replaces a single character, located in the same place.
 - = *RegEx*: Checks if the text matches a regular expression, returning 1 (*true*) or 0 (*false*). A regular expression is a sequence of characters that forms a search pattern. It is a standardized format whose most used expressions are the following:
 - *^*: Start of expression. Ex: */^Ba/* Any text that begins with "Ba".
 - *\$*: End of expression. Ex: */on\$/* All texts ending with "on".
 - ***: Matches the preceding character zero or more times. Ex: */ab*/* will admit "a", "ab", " abbb".
 - *+*: Matches the preceding character one or more times. Ex: */ab+/* will admit "ab", "abb", but not "a".
 - *?*: Matches the preceding character zero or once. Ex: */ab?/* will admit "a" or "ab".
 - *.*: Any character, except line break.
 - *SubStr*: Extract a part of the text, from the position *StartIndex*, length *Count* characters.
 - *Delete*: Delete a part of the text.
 - *RightStr / LeftStr*: Returns the ending / starting segment of text of specified length.
 - *UpperCase / LowerCase / Capitalize*: Converts text to upper / lower case.
 - *Length*: Returns the length of the text in characters.
 - *Trim*: Removes white spaces at the beginning and end of the text.
 - *Same*: Compares the text to a given value, returning *1-True / 0-False*.
 - *Split*: Segments the text delimited by the character entered and returns the segment that corresponds to the indicated number.
 - *After, Before*: Returns the text after or before the specified separator.
 - *Replace*: Enter the text you want to change along with the text you want to replace it with, separated with commas.
 - *Extract number*: Finds and gets the first numeric value in the text.
- **Function:**
 - *f(x)*: Executes a more complex operation, applying a polynomial function using "x" to represent the value read. It also allows you to use substitution parameters to refer to other variables. If the variable is writable, you will need to configure the reverse conversion function for this operation.
 - *Point curve*: The result value is obtained by linear interpolation from a predefined curve using a list of points (X,Y). You must parameterize the curve in the corresponding section of the *Notify General Settings* (19.11.1), and here only indicate the name of the curve to be applied.

6.3.4 Acquisition > Value mapping

Allows you to assign a value or a text to the different values that a variable can return. You can specify a specific value, a range using the starting and ending values separated by semicolons (for example "2;10"), or you can use the letter "x" to refer to the current value entering a polynomial function $f(x)$.

Note that depending on the type of transformation or mapping, the type of data may vary. That is, if you have a numeric variable and the value 1 is mapped as a text "ON", the final variable will not be numeric, but will become text. The concept of mapping must be distinguished from that of visualization. The first can affect the data itself, the second only affects the way the value is displayed.

6.3.5 Transfer

It allows transferring the value of this variable to another. Every time the read value changes, a writing will be made to the transfer destination variable. In this way, it is possible to interconnect different data sources.

The transfer can be made to another local variable, or to a remote variable in another notify instance, via rest API.

6.3.6 Actions

If this functionality is available in the license, it allows you to assign actions that will run whenever the value changes, or when there is an activation or deactivation. A value change from 0 to distinct of 0 is considered activation and vice versa for deactivation.

6.3.7 Data log

- **Enable - Internal (Notify):** Activates the data recording of the variable in Notify's internal database. Internal graphs on Notify read data from this database.
- **Enable - SQL:** If available in your license, it activates the data log in an external database type RDBMS (also called SQL), configured in the main *Settings*. In this case, you can specify an external ID associated with this variable to be used in the SQL of the external database if needed.
- **Enable - InfluxDB:** Activates the data log in the external InfluxDB database. In this case, a new tab is enabled to configure:
 - *Reg. Type:* Select the type of record to apply, from among those configured globally. If no custom record type is specified, only the standard Notify historian will be activated.
 - *Fixed tags:* Define the value for each configured fixed tag.
 - *Other tags:* Indicate, if necessary, other tags for this variable, in the format of InfluxDB Line Protocol: tag1=value1, tag2=value2...

Regardless of the destination of the historical data (internal / external), the control method and parameter configuration are common to all of them, with two main methods:

- **On change:** The value is saved when it changes, being possible to adjust these parameters:
 - **Deadband:** Required. Minimum variation the value must have, with respect to the last saved value, to cause a new record in the database. This setting is not available for

digital variables, which are always recorded at every change of state. By default, deadband is "0", which means that all value changes will be recorded.

- **Min. time:** Optional. It is the minimum time that must pass since the last sample for a new value to be recorded in the database. It is useful for filtering on variables whose value changes very quickly or continuously. It complements the "deadband" parameter, that is, for a new record to be saved, both conditions must be met.



Pay attention to the configuration of the historian parameters (dead band and times), especially for variables of rapid and continuous change, to avoid an overload of the process and the database.

Note that these parameters affect both the internal and external data log.

- **On time / Regular:** Data recording occurs at a regular interval. It supports simultaneous recording on change.

6.3.8 Bindings

You can dynamically link some properties to the state of another variable:

- **Writeable:** If the variable supports and is set to "Read/Write" mode, write will actually be allowed at run time if the bound variable is active (either *True* or non-zero value).

6.3.9 Data source types

Specific configuration depending on the protocol or type of variable:

6.3.9.1 API

- **Variable:** Configure the variable according to the selected API. The necessary parameters vary depending on the API, so we recommend going to the documentation of the same to have more information about the meaning and values necessary for each of them.

6.3.9.2 Database

The SQL query is defined at the reading group level, and it can return a single record or multiple records. To define the SQL, go to the reading group settings. Next, you need to configure for each variable in the group:

- **Field:** Enter the name of the field, or its position (starting at 1).
- **Record num. / Index field value:** Enter then record number for read the data. By default, the first one. If an index field is specified in the reading group settings, then the value for that field can be specified here to filter the records, instead the number.
- **Data type:** Indicate the type of source data.
- **SQL for writing:** If you want to allow writing in the variable, a specific SQL statement must be entered to perform the action, since Notify is not able to automatically obtain the possible INSERT or UPDATE required from a generic SELECT. When configuring this SQL more dynamically, the available parameters and macros can be used, being *":value"* the most important since it refers to the value to be written.

6.3.9.3 Calculated data

This data source includes operations on one or more variables or alarms. Also, for ease of implementation, other auxiliary variables and constants.

In the **Selection** field it is indicated on which variables or alarms the specific calculation will be carried out:

- **Variables:** Free selection | Sorted list | Data client | Group | Folder
- **Alarms:** Free selection | Folder
- **Free function**
- **Auxiliary variable:** Auxiliary variable | Constant | Timetable | Pulse train

In the **Operation** field, you must select what type of calculation to perform. The list of options depends on the selection made in the previous point but in general they are as follows:

- **Sum:** Returns the sum of all correctly readable variables.
- **Maximum | Minimum | Amplitude (max value – min value)**
- **Average**
- **Variance | Standard deviation**
- **Moving average:** Applies to a single variable. The result is the average value in the last time interval indicated.
- **Trend:** Returns the trend of the value of the variable in value and sign, calculated on the last values in the indicated time interval, and projected on a specific unit of time (second, minute, hour, day). Useful for determining whether a variable tends to grow (positive result), decrease (negative), or stabilize (trend will be close to zero).
- **Delay:** Returns the same value of the source variable, but delayed in time. Note that the maximum cached samples is limited to 3600, so the maximum delay time is also limited.
- **AND:** Returns 1 if all variables have a non-zero value, or if all alarms are active. 0 otherwise.
- **OR:** Returns 1 if any variable has a non-zero value, or if any alarm is active. 0 otherwise.
- **Count:** Returns the number of variables or alarms that matches the given criteria.
- **Activation counter:** It implements a counter of activations of one or more alarms or variables. An alarm activation is considered when it goes from an inactive state to an active state, excluding unknown state or reading errors. For a variable, the change from value 0 to any value other than zero is considered activation. The counter can be reset automatically every hour, day, month, or year. It can also keep the value indefinitely or be reset manually since this type of variable allows writing.



A variable of type "Activation counter" is useful when an alarm needs to be triggered with the number of occurrences of an event in a time interval. For example, it may be that the starting of a motor is a normal event, but if it starts more than 10 times in one hour it is considered an anomaly.

- **Cadence:** It measures the number of activations or pulses per unit of time. A "reset" time must be indicated from which, if there are no changes, the cadence value goes to zero.

- *Time between pulses (sec.):* Measures the elapsed time between two consecutive activations.
- *Active time (sec.):* Returns the time where at least one of the selected variables or alarms is active, that is, with a value other than zero. Note that value returned is in seconds; if you want another unit, you must apply a transformation.
- *Totalized:* Gets from an instantaneous data, the totalized in time. It is necessary to indicate the unit of time of the base variable. For example, from an instantaneous flow in "m3/h", it is possible to obtain the cumulative totalization in "m3", in this case indicating unit of time "h | Hour".
- *Copy:* Copies the value as is from the first selected variable.
- *First good value:* Copies the value of the first variable whose reading has good quality from among those selected.
- *User function:* It allows more specific or sophisticated mathematical operations to be executed, adding the necessary variables or alarms either manually using substitution parameters or by selecting the elements in the list of variables or alarms accessible through the "V" or "A" buttons, respectively. You must indicate what type of data is expected to be obtained as a result of the function.
- *Auxiliary variable:* This variable is not read from any source. It is a variable in memory, which can be used as a container for writing from an action or help for other operations. The value of these variables is retentive, that is, it is kept when restarting Notify.
- *Constant:* It is also an auxiliary variable, but unlike this one it cannot be modified or written. The value is saved directly in the variable settings and not in the state. Useful for use in other calculations.
- *Timetable:* Special type of auxiliary variable, which takes its value from a calendar configured at the global level (*Global Settings > Transformations > Timetables*). This can be of two types:
 - *Periods:* The value of the variable will be the current period number.
 - *On/Off:* The value will be True (1) / False (0) depending on whether in an active period or not.
- *Pulse train:* Special type of auxiliary variable, of digital type, which continuously alternates between *True/On/1* and *False/Off/0* according to configurable times.
- *Conversion:* From several variables in an ordered list, it allows you to get a result such as date, or convert ASCII characters to text.
- *Concatenate:* Gets a single result by putting together an ordered list of variables, which can be texts or bits.

6.3.9.4 Internal data

The variable returns an internal Notify data. Select the desired family and variable from the extensive list available. Names are descriptive.

6.3.9.5 Ethernet/IP

- **Tag:** The name of the variable on the device. Manually type the name or select from the browser if you have an online connection with the device and it supports this functionality.
- **Data type:** Type of data that corresponds to the tag entered.

6.3.9.6 *Historical in DB*

For source type "*Internal Data (Notify)*":

- **Variable / Alarm:** You can select a variable or an alarm. In case of alarms, the data is treated mathematically as digital variables, with 0 for inactive and 1 for active status.

For source type "*RDBMS / SQL*", SQL queries are mounted dynamically, based on this configurable data:

- **Table / View:** Name of the table or view to query.
- **Date field, Value field:** Name of fields to read date and value.
- **ID field, ID value:** Optional. Field name and value of the data identifier in the database.

For source type "*InfluxDB*":

- **Measurement:** Select or enter the measurement name, and optionally the retention policy.
- **Field:** Field name. Optional, default "value".
- **Tags:** Dynamically configure the tags that identify the series to be read. You can type manually or select from the drop-down.

6.3.9.7 *Microcom Zeus*

- **Channel:** Configure the channel number of the variable to read or select it from the list.

6.3.9.8 *Modbus*

- **Block:** Select the Modbus data block from which to acquire data:
 - *Coils* (coils or digital outputs; read and write)
 - *Discrete Inputs* (digital inputs; read only)
 - *Input Registers* (input records; read only)
 - *Holding Registers* (retention or general-purpose registers; read and write)
- **Register number:** Specify the record number to read, depending on the numbering style setting at the data client level (0..N-1 / 1..N).
- **Data type:** For digital data blocks, the data type is always binary (Bit / Boolean). However, for register-type blocks, although at protocol level the readings are made by words, it is possible to select the type of data to process:
 - *Bit / Boolean:* In this case, indicate the number of bit to read, from 0 to 15, with 0 being the least significant.
 - *Byte / UInt8:* Takes the high (left) or low (right) byte of the word.
 - *Word / UInt16:* Default data type in Modbus.
 - *DWord / UInt32:* Double word, or 32-bit unsigned integer. The Big Endian or Little Endian configuration at the server level is considered when decoding the value.
 - *Integer (8, 16, 32 or 64 bits):* Signed integer.
 - *Float (32 or 64 bit):* Floating point, one or two words.

6.3.9.9 OPC

- **OPC item:** Name of the item on the OPC server. Usually you will not have to modify it manually, since you can navigate on the OPC server browser and select it. For your information, you will see the original data type, as well as the length in bits, if available.

6.3.9.10 Network

- **Data:** Select the data to obtain:
 - *Ping:* Binary value (0/1) indicating whether a specific computer (host) responds to ping requests or not.
 - *Ping (ms):* The value read is a 16-bit integer, indicating the response time of a ping request.
 - *Internet:* Binary value (0/1) indicating if there is an Internet connection available.
 - *TCP Port:* Binary value (0/1) indicating whether a TCP port is open or not.
 - *FTP:* Binary value (0/1) indicating whether an FTP port is open or not.
 - *FTP login:* Binary value (0/1) indicating if an FTP connection with username and password is available or not.
- Configure the required parameters requested, depending on the type of data to be acquired (host, port, timeout ...)

6.3.9.11 Siemens S7

- **Address:** Indicate the address of data to be read. You can type it in the text field or use the quick selection combos.

The syntax of the address depends on the memory area and the type of data:

- **Areas:**
 - Digital inputs (I, E)
 - Digital outputs (Q, A)
 - Markers (M, F): Marks or flags.
 - Timers (T)
 - Counters (C, Z)
 - DB (DB): Data blocks.
 - Variable Memory (VM, V)
 - System: System data, CLOCK, which allows reading and writing the PLC. It is not accessible in all models. If you try to read from a PLC in which the value field is not available, it will get an error.
- **Data type:**
 - Booleans: Bool [X] (1 bit)
 - Integers: Byte / USInt [B] (8 bits), Sint [SB] (8 bits), Int [SW] (16 bits), Word / UInt [W] (16 bits), DWord / UInt [DW] (32 bits) , DInt [SD] (32-bit), LWord / UInt [LW] (64-bit), LInt [SL] (64-bit)
 - Real: Real [F] (32-bit), L Real [LF] (64-bit)

- *Text*: Char [C] (8 bits), String [ST], S7String [SST]
- *Date and / or Time*: Time [TM] (32-bit), Date [DT] (16-bit), Date_Time [DTL] (64-bit), Time_Of_Day [TD] (32-bit), S5Time [T] (16-bit), LTime [LT] (16 bit), L_Time_Of_Day [LTD] (64 bit)

For inputs (I), outputs (Q), flags (M) and variable memory (VM) the syntax is:

<Area><Address> Example: IBO, QF2, MWO, VDW5

For counters and timers:

<Area><Index> Example: T2, C4

And finally, for data blocks it would be:

<Area><Block No.>.DB<Address> Example: DB1.DBW3

The *Address* field follows different syntax depending on the data type. If it is a Boolean: **X<Byte>. <Bit>** (Example: DB14.DBX1.1).

In the case of a text string: **[S]ST<Initial_Byte>.<Length>** (Example: DB2.DBST0.5), where *Length* is the number of bytes that make up the string.

While the rest of the formats adapt to the following pattern: **<Format>< Initial_Byte>** (Example: DB20.DBW1).

6.3.9.12 File system

- **File / Directory**: First, select if the data to be acquired concerns a file or a directory.
- **Path**: Indicate the absolute or relative path of the file or directory to check. If a base path has been configured at the reading group level, the path entered here will be relative to that one.
- **Data**: Select the data you want to check:
 - *Exists*: Binary value (0/1) indicating if the file or directory exists.
 - *Size*: 64-bit integer, indicating the size in bytes of the file or directory.
 - *Total files*: In the case of a directory, it indicates the number of files whose name matches the given mask.
 - *Created, Last access, Modified*: 32-bit integer, which indicates the time elapsed in seconds since the creation, access or modification of the file or directory.
- **Mask**: In the case of a directory, and certain data, optionally a name mask can be indicated to filter files, using "*" to mask parts of the name or extension. By default, the blank mask is equivalent to a "*.*"
- **Recursive**: In the case of a directory, and certain data, it indicates if search for files only in the root folder or in all the folders it contains.

6.3.9.13 OS

- **Data**: Select the data to monitor:
 - *Services and processes*:
 - *Service running*: Digital value (0/1) indicating if a given Windows service is running. Enter the internal name of the service.

- *Process running*: Digital value (0/1) indicating if a given process or application is running. Enter the name of the executable, with or without full path. If with path, the execution of that specific process is controlled. If only the executable name is specified, it controls that any process with that name is running, independent of executable location.
- *Disk**:
 - *Available space*
 - *Used space*
 - *Total space*
- *CPU*:
 - *Total*: Gets the total percentage of CPU used.
 - *Application*: Gets the percentage of CPU used by Notify.
- *Memory**:
 - *Total*
 - *Total in use*
 - *Total available*
 - *In use by application*
- *Battery*: Only available on systems with a battery.
 - *Battery mode*: Digital value (0/1) that indicates if the equipment works in battery mode.
 - *Charge percent*: Analog value (0..100) that represents the % of charge remaining.
 - *Time left**: Battery remaining time value, if available.
- *Times**:
 - *Time since system boot*
 - *Time since application start*
 - *Time since user interaction*: Time elapsed since the user has used the mouse or keyboard.
 - *User interaction*: Digital variable (0/1) that indicates that there has been user interaction (mouse or keyboard) during the last N seconds.
- *Clock*:
 - *Current time*: Returns the current date and time. Use the date transformation functions, available on *Acquisition* tab, to extract specific data if necessary.



**For measurements of disk or memory space, the default unit is the byte, but Notify automatically interprets the configured unit (% , KB, MB, GB, TB...), converting the value read to that unit.*

For time measurements, the default unit is seconds, but units are also interpreted and the value is converted to typical units (s, sec, seconds, m, min, minutes, h, hours, d, days, w, weeks, mo, months, y, years ...)

6.3.9.14 Telegraf (InfluxData)

- **Filter tags:** Indicate the *tags* and values to filter the data and obtain the value corresponding to the variable. If the Telegraf agent is configured and operational, you can select from the list of received tags. If not, you must indicate them manually.
- **Field:** Indicate the name of the field to read.
- **Data type:** Configure the type of data that Telegraf returns, to be able to treat the value read.

6.3.10 Editing

- **Direct editing in variable list:** Some text fields, such as name, OPC item, aliases, units, etc., can be modified directly on the list of variables. Select the column to edit and press F2 or slow double click.
- **Drag / Drop:** You can move variables from one group or folder to another by selecting them from the list and then dragging and dropping on the destination to which you want to move them. You can also move entire folders in *Folders* view.
- **Copy / Paste:** Notify allows you to copy and paste one or more variables, which is useful for creating variables that are similar to other. After pasting the new variable, you can edit the differences as needed. It is also possible to copy and paste entire folders.
- **Search for:** It allows locating variables by searching for a text in the name field, aliases, etc. The search is performed only on the currently selected variable list. There are two search functions: A quick search (F7, field in the top right corner of the window), which simply filters the results that contain the text; And a complete one (Ctrl + F or contextual menu), which locates the occurrences of a text in the list, iteratively (F3), and which allows more complex criteria such as upper / lower case, forward / backward, etc.
- **Replace text:** In a multiple selection.
- **Multiple editing:** If you select several variables in the list, in the context menu (right button), under the "Modify" menu you will find multiple editing methods, such as changing units, number of decimal places, copying display properties or transformations from one variable to others, etc. .
- **Column configuration:** If you right-click on the header of the variable list, you can select which fields you want to show. It also has options for a quick adjustment of the width of the columns, which adapts to the current content and the available display width.
- **External editing using CSV / Excel file:**

When you have to edit a large list of variables, you may find it more convenient to use an external tool that allows you to edit files in CSV format, or directly Microsoft Excel®, in which you have powerful utilities to facilitate mass editing.

To facilitate this process, Notify allows you to export and then import the list of variables from a CSV or Excel file. Use the "Export to File" button on the top button, and save the file where you can edit it, or leave the file name blank to create a temporary file. When you export existing variables, you can select which, as well as the list or arrangement of columns or fields (all, basic list, current view, or custom list).

This procedure allows you to edit **existing variables**, which are linked by the ID field, and add **new variables** indicating a specific ID that does not exist, or leaving it blank for Notify to generate automatically.

Use the external tool that is most appropriate for you. For CSV, Notify allows you to use both the comma and semicolon to separate the data, as there are tools that work best with one format or another. You can change this parameter in the General *Settings* window.

To then integrate the changes into Notify, use the "*Import File*" button on the top button. When importing a file, Notify can interpret the name of the column in the header, or it can indicate a specific layout using a custom list. For an Excel file, it is possible to automate repetitions by letting Notify change the value of a specific cell in sequence. If you use linked formulas in Excel you will see that it is useful for creating repetitive variable templates.

If incorrect or invalid values are found when processing the data, a window will be displayed with the incidents found. Check the syntax if necessary, and in any case verify that the final configuration is the desired one.



Use external editing using CSV or Excel when you need to add or modify a large list of variables, or several variables that are similar to each other.

6.3.11 One variable alarms

Depending on the selection of variables, the list of alarms configured on them is shown in the lower panel, with the main parameters and the current status.

From this list you can access and modify the detailed alarm settings by double clicking on it. It also has a direct access to the historical graph of the alarm.

Although the detailed editing of alarms can be done from the main Alarms section, it is also possible to add, configure or delete alarms from this panel on a specific variable, or to quickly create typical alarms, by selecting several variables. The next chapter will cover alarm settings in detail.



Use the lower Alarms panel, in the Variables editing window, for quick editing of alarms on a certain variable, while the full Alarms window that will be described in the next chapter is more appropriate for massive or more advanced editing.

6.3.12 Simulate value

In the popup menu clicking with the right mouse button on a variable you will find the option "*Simulate value*". Allows you to temporarily change the value of the variable. This is useful for debugging and testing purposes.

When the value of the variable is simulated, a symbol with a red hand is shown. The simulation will not end until the contextual menu "*End simulate value*" is clicked.

7 Alarms

In Notify, the definition of variables and alarms are independent. This allows one or more alarms to be declared on the same variable and facilitates the configuration of multiple similar alarms on different variables.

Nombre	Alias	Carpeta	Prioridad	Variable	Valor actual	Condición	Política	Estado	Ack	Fecha activación	Tiempo activa
Alarma de valor alto			Normal	Bucket Brigade.Int1	0	> 10	Alarmas prioritarias	Inactiva		-	-
Datos - Error en servidor		Sistema	Normal			?	-	Deshabilitada		-	-
Datos - Error lectura alguna variable		Sistema	Normal			?	-	Deshabilitada		-	-
Email - Error en sistema de envío		Sistema	Normal			?	-	Inactiva		-	-
Email - Límite de mensajes		Sistema	Normal			?	-	Inactiva		-	-
Error de registro en BD externa		Sistema	Normal			?	-	Inactiva		-	-
GSM - Error en sistema de envío		Sistema	Normal			?	-	Deshabilitada		-	-
GSM - Límite de mensajes		Sistema	Normal			?	-	Deshabilitada		-	-
Notify - Error interno, consultar log		Sistema	Normal			?	-	Inactiva		-	-
Notify - Informe de estado		Sistema	Evento			?	Sistema	Sin inicializar		-	-
Notify - Inicio aplicación		Sistema	Evento			?	-	Inactiva		-	-
Notify - Nueva versión disponible		Sistema	Evento			?	-	Deshabilitada		-	-
Notify - Sin acceso a Internet		Sistema	Normal			?	-	Deshabilitada		-	-
Pasarela SMS (Esendex) - Error en sistema de envío		Sistema	Normal			?	-	Deshabilitada		-	-
Pasarela SMS (Esendex) - Límite de mensajes		Sistema	Normal			?	-	Deshabilitada		-	-

Illustration 22: Alarms section.

7.1 System alarms

Notify has a set of predefined system alarms, which can be registered or notified in the same way as any other user alarm. In edit mode you can change some properties, depending on the specific alarm, as well as disable them if they are not useful to you.

- **Data – Data client fail:** It is activated when a data server could not be connected. In this case, check the log files for more information, check data client settings, and verify that the server is available.
- **Data – Read error in some variable:** It is activated when connection to the data server is ok, but there is a problem when reading a variable. In this case, find in the list of variables which one or which are causing this problem and check settings and variable is available on the server.
- **Data – Write error to some variable:** Indicates that a write action, local or remote, could not be successfully executed. Locate in the general log the details of the incident in case something needs to be corrected.
- **Email / GSM / SMS Gateway - Notification system failure:** There is an alarm of this type for each enabled messaging system, and it is activated in case of incidents when sending or receiving messages. If active, check the log files to locate the problem, and take action accordingly if necessary.
- **Email / GSM / SMS Gateway - Message limit:** For each messaging system you can configure a message limit, either daily, monthly or both. Once reached that volume of messages sent, Notify stops using that system, and this alarm is triggered.

- **Error in data log to external DB:** Indicates that there have been incidents saving data to an external database (SQL or InfluxDB). Review the specific log to locate and resolve the issue.
- **Notify - Internal error, see log files:** If an unexpected error occurs that may cause a crash in the application, it is captured and described in the log files, activating this alarm to indicate that it should be reviewed, since it does not show any message on the screen.
- **Notify – State report:** It is not an alarm, and it will never be active, but the same settings are used to allow assigning which contacts will receive a message with the periodic status report if this is enabled.
- **Notify – Application started:** It is also not an alarm, but an event, which is activated for a few seconds right after launching the application. It allows to notify this event to a contact, or to quickly view in the alarm graph when the computer or the application has been restarted or active.
- **Notify - New version available:** If you have an Internet connection, this will notify you if a newer version of the application is available for update.
- **Notify - No internet access:** Activates when the Internet connection is not available. It can be useful to notify by SMS if you need to monitor this circumstance.

7.2 Project alarms

We refer to project alarms as those created by the user, so you can see them also referenced as user alarms in this manual or in the application.

An alarm is essentially a declaration of the variable to be controlled, a condition and an alarm value, which is assigned a descriptive name that usually refers to the name of the variable itself. With that little data, you would already have an alarm defined in Notify. However, we will see in the following sections that there are many more parameters for a fine adjustment.

7.3 Query and visualization

The alarm management window is very similar to the one previously seen for managing variables (see *Illustration 5*). The navigation tree also has several views to organize the alarms. Some of them are designed to help the configuration quickly and visually, being the *Folders* view the most common and the default view. In the folder tree, the two root nodes are fixed, and they separate the system and the project alarms. Under project alarms, the number or nested folders can be unlimited. There are no special settings or features associated with folders. Folder name will be usually inherited from the variable and used to separate alarms by zones or different system processes.

There are two modes to query the alarms: *Summary* and *Historian*.

7.3.1 Summary

In *Summary* mode, the current status of the alarms is displayed. You can choose between three different views:

- **All:** Shows the current status of all alarms.
- **Enabled:** Shows all enabled alarms, regardless of whether they are active or not.
- **Active:** It is the default mode. Only the alarms active now are displayed.
- **Unacknowledged:** All alarms pending acknowledgment are displayed, still active or not.

In any case, only the alarms belonging to the folders selected in the side tree are shown, or all if none is selected.

In the status bar you will see a series of counters, with relevant information such as the total number of alarms declared, how many are active, or the maximum limit allowed by your license.

The main list shows information on the alarm, current status, or activation date if applicable. Some notes to mention:

- **Icon:** Graphically displays the current status of the alarm. The icon depends on the alarm status, priority, and time.
- **Value:** Shows the current value of the main variable that is controlled in the alarm.
- **Priority:** In addition to the text, an icon is displayed with the color of the alarm priority, if it is active. If it is not active, the icon is always gray.
- **Status:** Displays the current status of the alarm in text:
 - *Not initialized:* The current state has not been acquired for the first time yet.
 - *Unknown:* Unable to know the alarm status. It can occur if the base value of the variable is not known.
 - *Disabled:* Either the alarm or the base variable is disabled by the user.
 - *Hour disabled:* The current time is outside the alarm control hours.
 - *Inactive / Active*
 - *Inactive / Active, waiting stabilization ...:* The alarm condition is met, but the time configured as value stabilization has not yet passed, after which it will become inactive/active. The pending time is indicated.

7.3.2 Historian

In *Historian* mode you can query the alarm activation and deactivation event log between two given dates. It has a quick date selector (today, yesterday, this week, etc.) or you can select two dates freely. In this mode, in addition to folders, it is possible to filter by a specific alarm by selecting it in the side tree. You can export the displayed data to a file, using the *Save* option that you will find in the popup menu (right mouse button).

7.4 Editing and configuration

As in the declaration of variables, the Edit mode is accessed through the "*Enable editing*" button.

▪ Alarm settings (Add or Edit)

Adding new alarms or modifying existing ones consists of configuring the following fields.

- **ON / OFF:** Allows you to disable the control of the selected alarm without having to delete it.
- **Folder:** Write the path of the folder where to locate the alarm in the tree, using the separator "/" for subfolders, or inherits this path from the variable, linking it with the button located to the right of the field.
- **Name:** Text that identifies the alarm, and that will be used in visualization and notifications. You can link the name of the alarm to the name of the main variable being monitored, with the buttons on the right of the field. If you enable this link, you have three additional quick

configuration options, which allow you to append a globally configurable prefix for typical cases of high value (*Hi*), low value (*Lo*), out of range value (*Out*), or fail (*Fail*).

- **Alias:** Optionally you can indicate a short text that identifies the alarm, which can be used in SMS messages or queries.
- **Priority:** Allows “cataloguing” alarms according to their severity, assigning a value between *Event* (blue), *Low* (purple), *Normal* (yellow), *High* (orange), or *Critical* (red). The priority will then allow you to discriminate in some configuration parameters, or filter alarm lists.

Alarm condition:

- **Variable:** Select the main variable on which this alarm is controlled. Use the selection list (F9) since it does not allow manual typing.
- **Condition:** Select the criteria from among the available ones that suits your needs:
 - $=, <>, >, > =, <, <=$ → There is an alarm if the value of the variable is equal, different, greater, etc., than the one indicated. The comparison conditions ($>, > =, <, <=$) allow setting a hysteresis value, to be considered to detect deactivation. Thus, for example, if an alarm is configured with a value > 100 , and hysteresis = 20, the alarm will activate when it exceeds 100 and it will deactivate when it returns below 80. If the condition is < 100 , also with hysteresis = 20, it will activate when falling below 100, and deactivate when returning above 120.
 - *Bit = 1, Bit = 0* → There is an alarm if in the base variable the indicated bit number is 1 or 0, respectively. Being the bit number 0 the least significant bit.
 - *Out, In* → Two values are configured, and there is an alarm if the value of the variable is outside ($< min$ OR $> max$) or inside ($>= min$ AND $<= max$) of the indicated range.
 - *It does not change* → A time is configured, and there is an alarm if the value of the variable does not change during the indicated time.
 - *Null* → There is an alarm if the value of the variable is Null.
- **Value / Variable / Count:** In the usual case of using a comparison operator (first point in the previous list), you can indicate a fixed value or compare against the current value of another variable to control that there is an alarm. If the base variable is of the counter type, it is possible to define alarms based on the count value for a certain time.
- **Condition 2:** Optionally, a second condition can be configured. The alarm will be active when both conditions are met simultaneously. This second condition can be the value of another variable or another alarm.

Stabilization:

- **Stabilization time:** To avoid false activations in data that fluctuates too much, Notify allows you to configure a waiting time from activation or deactivation to consider the status change effective.

Inhibit other alarms:

Allows you to temporarily disable the notification of other alarms of equal or lower priority, which are considered dependent on it. That is, while this alarm is active, no subsequent activations/deactivations of the inhibited alarm list will be notified. A useful example can be a voltage failure alarm, which causes failures in several devices. In this case it may be interesting to receive only the general voltage failure, and not the list of all individual failures. The option of which alarms to inhibit depending on the location folders can be: In the same folder / On the whole branch line / All.

Hours:

Configure if you want one or more time bands for which the alarm is to be controlled. By default, they are created on a continuous 24-7 schedule. Note that outside of the indicated hours, the alarm is considered disabled, not inactive. So, when leaving or entering the controlled time, it does not generate activation or deactivation events unless it has changed state with respect to the last known one.

Message:

- **Add to message...:** Select, if desired, data that you want to add in the alarm notification message. It is possible to add:
 - Another variable, different from the alarm one.
 - A user query, among those defined in the "Queries" section. In this case, the value or status of several variables or alarms will be added to the message.
 - A system query.



You should be careful when configuring the message template, as well as the associated information, especially in the case of SMS, since the content of the resulting message could be too long, causing an extra cost due to the consumption of several SMS or the truncation of the text to the maximum permissible length.

- **Activation / Deactivation:** Select how you want to be notified of these status changes. By default, both will be notified with the global template, but by checking the option "Custom text" you can create a different message text for that specific alarm.
- **Acknowledge:** Set the incoming message format for alarm acknowledgment. For this you can use:
 - *Global template:* If an alarm needs acknowledgment, the notification message will contain its ID by a number preceded by "#". To acknowledge the alarm, the recipient must respond with a message that contains that identifier and begins with: **OK**, **ACK** or **REC**, with a space before the alarm identifier. It also recognizes the "RE:" prefix, so that for an email notification it will be enough to simply reply to the message, since the mail client will add that prefix in front of the response and the ID goes in the subject itself.
 - *Text:* Enter a specific text that will acknowledge the alarm.
 - *MS-DOS pattern:* Indicate an MS-DOS pattern to which the message will be compared and if it is true the alarm will be acknowledged.
 - *RegEx pattern:* Enter a regular expression to identify the messages that will acknowledge the alarm.

The last three options include:

- *Test:* Allows you to enter a message and check if it would recognize the alarm or not.

- *Case sensitive*: Check this option if you want to be case sensitive when checking.
- **Standard email template mode**: Allows you to customize the HTML template used for standard alarm notification. You can select between the preset modes or your customized own:
 - *Reduced*: The notification contains only the text (message or name) and alarm dates.
 - *Normal*: Also include if available the folder, priority, and current value of the variable.
 - *Full*: All available fields.
 - *Custom*: Allows to select which fields to include. The custom template is common to all alarms, and is available in the *Global Settings*.

Notify to...:

- **Notification policy**: If needed, select the name of the notification policy to apply to this alarm. Remember that the notification policy determines which contacts it is sent to, whether it requires acknowledgment or not, etc.
- **Direct assignment**: Specify which contacts or team you want to notify specifically about the alarm.
- **Effective contacts**: Shows contacts that will actually be notified based on the current settings.

Actions:

Specify what action or actions will be executed, previously configured in the main *Actions* section, when the alarm is activated, deactivated and / or when it is acknowledged.

Data log:

Enable here the log in the external database, if available. The internal log in Notify is always enabled by default. The record is always triggered by change, but it is also possible to configure a fixed time recording as well. Beyond these differences, the configuration in this section is like the one already seen for a variable.

Bindings

You can dynamically link some properties to the state of another variable:

- **Enable**: At run time the alarm will be enabled if the bound variable is active (*True* or non-zero value).
- **Mute**: The linked variable controls the muted state of the alarm.

In the same way as in the editing of variables, you have these functions when editing alarms:

- **Direct editing in list**: Some text-type fields, such as name or alias, can be modified directly on the alarm list. Select the column to edit and press F2 or slow double click.
- **Drag / Drop**: You can move alarms from one folder to another by selecting them from the list and then dragging and dropping on the folder or group to which you want to move them. You can also move entire folders by dragging in the tree.
- **Copy / Paste**: Allows you to copy alarms or entire folders.
- **Search / Find**
- **Replace text**: In a multiple selection, it allows to replace part of a text field (name, alias). Useful when duplicating entire folders.

- **Multiple editing:** If you select multiple alarms in the list, in the context menu (right button), under the Modify menu you will find multiple editing methods, such as changing the alarm condition, stability times, contact assignment, enable / disable, etc.
- **Column configuration:** If you right-click on the header of the variable list, you can select which fields you want to show. It also has options for a quick adjustment of the width of the columns, which adapts to the current content and the available display width.
- **External editing via CSV / Excel file:** When you must edit a large list of alarms, you may find it more convenient to use an external tool that allows the editing of files in CSV format, such as Microsoft Excel®, which has powerful utilities to facilitate mass editing.

7.5 Acknowledgment of alarms

If the associated notification policy requires acknowledgment, each time the alarm is activated, it will go to the status pending acknowledgment. In this way, even if the alarm is deactivated, you can check all pending alarms in the *Unacknowledged* view.

There are several methods to acknowledge an alarm:

- **From the application:** The *Ack* column of the alarm list indicates whether it is acknowledged or not. Selecting the specific alarm, use the top *ACK* button or the equivalent option in the popup menu. From the *Unacknowledged* view, you also have the *ACK all* button. It is possible to configure the user level required to allow acknowledging alarms in the application, from the *Users > Permissions* section of the main *Settings*.
- **Message:** A contact, upon receiving the alarm notification, can acknowledge it by replying to the received message with the acknowledgment format specified for the alarm.
- **Missed call:** It is also possible to acknowledge alarms by a call. In this case, all those pending alarms assigned to the contact making the call will be acknowledged, if the contact was in active hours at the time the alarm was activated.

7.6 Mute alarms

Notify allows you to mute one or more alarms temporarily or permanently. Note that mute is different than disable alarms. Muting only disables the alarm notifications, while the control, status log, actions, etc., is still enabled.

You can mute one or more alarms using the corresponding option in the popup menu, or mute all using the button at the top right of the alarm window. You will be asked for the time during which to mute the alarms, being able to indicate "Undefined". In this case, you must remember to remove the mute mode manually.

The option to mute alarms can be useful during maintenance tasks, or to avoid well-known alarms that are regularly triggered or await correction.

7.7 Simulate activation

From the *Summary* alarms mode, selecting an alarm and clicking the right mouse button shows the option "*Simulate activation*" that will trigger the activation of the alarm. This is useful for debugging and testing purposes.

If the activation is simulated, a symbol with a red hand is shown. The simulation will not end until the "*End simulate activation*" contextual menu is clicked.

8 Notification policies

Notification policies determine how and to which contacts alarms will be reported. This centralized configuration is easier and more visual than direct assignment of contacts to alarms. Each alarm can have only one notification policy.

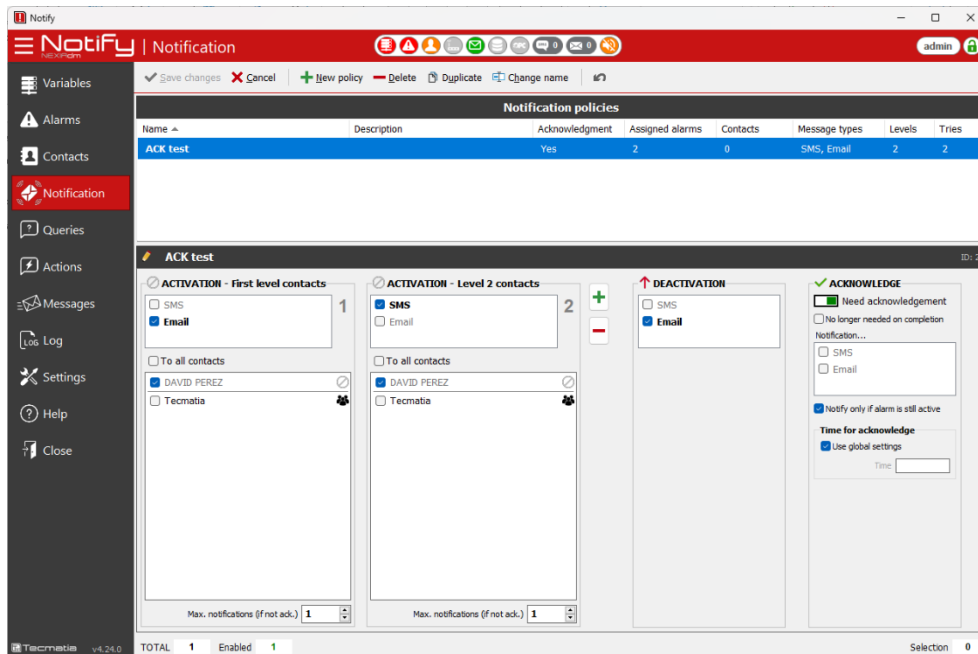


Illustration 23: Notification policies.

For each notification policy, the following settings are needed:

- **Name:** It will be used to assign to alarms.
- **Description:** Optional, more detailed text.
- **Activation:** Add one or more levels of contacts to notify of alarm activation. For each level, specify the message types, assigned contacts (or all), and maximum notification attempts before switching to the next level. The notification protocol for new alarms assigned to this policy is as follows:
 - When the alarm is activated, the type(s) of message is sent to the assigned and active contacts that allow that message type (SMS, email...).
 - If the acknowledge is enabled, and the alarm is not acknowledged within a configured waiting time, while it is still active, the contacts of the same level are notified again until the maximum number of attempts indicated (by default one).
 - Upon reaching this limit, after a new wait, the next level contacts are notified.
 - The waiting cycle for acknowledgment or deactivation / new notification continues until the total configured levels and attempts are reached.
- **Deactivation:** Activate the message types to be used to notify the deactivation of the alarm. The message will be sent to all contacts who have received the active alarm notification.
- **Acknowledge:** Indicate if alarms with this policy need acknowledgment. If yes, configure the message types to be used to notify the acknowledgment of the alarm. The message will be

sent to all contacts that have received the active alarm notification, including the same contact that acknowledge it. Note that if you do not enable acknowledgment, the activation or deactivation will be notified only once to direct or first-level contacts.

- **Time for acknowledge:** This is the amount of time Notify will wait for someone to acknowledge the alarm before notifying the next level of contacts. There is a global setting, configured according to the priority level of the alarm, or a specific time can be indicated for a specific policy.

9 Queries

If this feature is present in your license, Notify allows data consultation by sending an SMS or an email to the application. The result is always received on the same channel, and to the same individual sender.

The query texts, or commands, are not case-sensitive. In the application they will always appear in uppercase letters, to homogenize and be easier to identify.

It is possible to send several queries with a single message, separating them by commas. However, you should be careful with this in the case of SMS, since the result may be too long, and its reception and the number of SMS to be priced will depend on how you have configured the application, the operator's criteria or the conditions of your contract.



Query or Action messages are not case-sensitive.

You can send multiple queries or actions on the same message, separating them with commas.

9.1 Predefined system queries

Notify has by default a series of predefined query commands. The default commands are shown below, although you can customize them:

- **Application state (STATE):** Returns a message or status report, with the same format as the periodic status report, regardless of whether it has been activated or not.
- **Version information (VERSION):** Returns application version data.
- **Counters information (COUNTERS):** Returns information on the number of messages sent (day / month).
- **Disk space (DISK):** Information on the used and available space on the disk.
- **IP addresses (IP):** Returns the IP address of the computer where Notify is installed.
- **List of variables (VARIABLES):** Returns the complete list of variables and their current value. Not recommended for SMS.
- **List of alarms (ALARMS):** Returns the complete list of alarms and their status.
- **Active alarms (ACTIVE):** Returns the complete list of current active alarms.
- **Unacknowledged alarms (UNACK):** Returns the list of alarms pending acknowledgement.
- **Silenced alarms (SILENCED):** Returns the list of current silenced alarms.
- **System information (SYSTEM):** Returns data related to the system, such as operating system, start date, etc.
- **Screenshot (SCREEN):** Only for email, returns attached to the response email a file with a capture image of the screen.

You can modify these messages from "*Settings* → *Queries / Actions* → *System queries*". If you want to disable any of them from being available, just leave the text field blank.

9.2 User queries

A user query consists of a set of one or more variables and / or alarms, with a specific name, what are we going to call "command".

You can access the *Queries* section from the left side menu.

In the left panel you can see the list of all user queries defined. When you select one of them, you will see in the right pane the list of variables and / or alarms it contains.

9.2.1 Display

In addition to allowing the application to query data by sending a message, queries can be useful for quickly grouping and displaying a series of related variables or alarms, since in the right pane you will see the current value of the data in real time.

In view mode you have these options:

- **Preview:** It allows you to see the response message that the application will send, considering the current settings and data. If your license has several message formats (SMS, email ...) it shows a drop-down to select one of them.
- **Send to...:** Now send the reply message to the query to the recipient you select, from the list of contacts currently configured.
- **Graph:** In the right pane you have a direct access to the historical graphs component for the selected variable or alarm.

In the bottom bar of the window you can see the total number of configured queries, as well as the maximum available for your license.

9.2.2 Editing

As usual, you can access the edit mode by pressing the "*Enable editing*" button located in the header.

On the left side the list of defined queries is managed, being possible to organize them into folders using the Drag / Drop technique.

On the right side, the list of variables or alarms contained in the query is managed, being able to order them freely. It is also possible to assign a specific alias for each variable in this particular query by overwriting its long name or global alias.



When you modify the configuration of Queries, Actions or Aliases of variables and alerts, Notify checks and warns you if these are repeated, so that you can locate and correct these settings. It is recommended that there are no aliases or repeated commands, as it can lead to confusion and the result of queries or actions would be unpredictable, solving by the first occurrence it finds.

9.2.3 Automated queries

It is possible to automate the sending of queries. In the editing window of each query, you can activate this feature, indicating what type of messages to send, one or more schedules, and which contacts or teams. In addition, the option "*Automate*" is added to the popup menu (right mouse button on the list of queries). This option allows you to quickly enable or disable automation for a selection of

multiple queries. To facilitate visual identification, a clock icon is displayed in the list for each query that has automation enabled.

9.3 Direct query

You can obtain the value of any variable or the status of any alarm without having to configure a user query, simply by sending a message with the name or alias of the desired variable or alarm.

10 Actions

If this feature is activated in your license, Notify can execute “actions”, associated with the activation or deactivation of alarms or events, or by sending an SMS or email to the application. As it is through incoming messages, you can configure that an OK message is answered after performing the action, which will be notified through the same channel and to the same sender that requested it. In case of error when performing the action, this will always be notified, regardless of the configuration. It is possible to execute several actions associated with the same event, or with a single message, separating the commands by commas. They can even be combined with queries. Each command will be processed in order and individually.

10.1 Predefined system actions

Notify has by default a series of predefined action commands. The default commands are shown below, although you can modify them to your liking:

- **Restart Notify** (`RESTART`): Forces the application to restart.
- **Restart PC** (`RESTART PC`): Forces a complete restart of the PC. It produces the same effect as when the user selects the "Restart" option in the Windows start menu. Be careful when executing this action in a remote way, because if another application is blocked waiting for a user action, it could cause a crash and prevent a restart. Notify tries to force the restart, but the actual execution of the restart is always the responsibility of the operating system.
- **Enable contact** (`ENABLE CONTACT`): Activates alarm notification for the sending contact if it exists as a declared contact.
- **Disable contact** (`DISABLE CONTACT`): Disables alarm notification for the sending contact if it exists as a declared contact.
- **Enable alarm notification** (`ENABLE ALARMS`): Activates the alarm notification system for all contacts.
- **Disable alarm notification** (`DISABLE ALARMS`): Deactivate the alarm notification system for all contacts.
- **Silence alarms** (`SILENCE`): Mute all alarms.
- **Cancel silence alarms** (`CANCEL SILENCE`): Cancel the silence of all alarms.
- **Enable SMS notification** (`ENABLE SMS`): Activates the sending of SMS (affects alarms, queries, writings ...)
- **Disable SMS notification** (`DISABLE SMS`): Disables the sending of SMS (affects alarms, queries, writings ...)
- **Enable email notification** (`ENABLE EMAIL`): Activates the sending of emails (affects alarms, queries, writings ...)
- **Disable email notification** (`DISABLE EMAIL`): Disables the sending of emails (affects alarms, queries, writings ...)
- **Enable state message** (`ENABLE STATE`): Activates the sending of the periodic status report message (affects all contacts assigned to it).
- **Disable state message** (`DISABLE STATE`): Deactivates the sending of the periodic status report message (affects all contacts assigned to it).

You can modify these default messages by going to “*Settings* → *Queries / Actions* → *System actions*”. If you want to disable any of them from being available, just leave the text field blank.

10.2 User Actions

If you click on the “*Enable editing*” button, you will go to edit mode, which allows you to add, modify or delete commands and actions. For each one you must configure these fields:

- **Name / Command:** Unique short text with which the action can be identified and executed.
- **Description:** Free text. Optional.
- **Manual execution:** Allows whether the action can be executed manually from the application, and the security level with which it can be executed.
- **Remote execution (API / incoming message):** Enables the execution by sending a message to the application. Allows you to select how the message should be to execute the action:
 - *With name or command:* Check this box if you want to execute the action by sending a message with the name of the command. It is the default mode.
 - *With text that match pattern ...:* Check this option if you want to create a pattern (Text, MS-DOS or RegEx) with which to compare the incoming message. If the comparison is matched, the action would be executed.

Each command can contain one or more actions, of the same or different types. In the right pane, each command will be assigned the actions that will be executed when it is triggered. The actions that belong to a command are executed in the declared order. Each action can optionally specify a condition that will be previously evaluated to determine whether to execute or skip and move on to the next one. The supported action types are detailed below.

10.2.1 Writing

It allows modifying the value of a variable that allows writing. To configure a write, you must select the variable and indicate the value to write. You can specify a specific value, a function $f(x)$ using the “x” to refer to the current value of the variable or use substitution parameters to reference the value of other variables.

10.2.2 Execute application

It allows the execution of commands or applications external to Notify, indicating the type of operation (typically ‘open’), show mode, executable file, parameters and working directory, according to the parameters necessary for the Microsoft Windows *ShellExecute* function. Being a simple action, it is very powerful since it allows the execution of practically any command, opening the possibility of expanding Notify functionalities or interconnection with other systems.



Using external “Execute application” actions, you can expand Notify's functionalities or interact with other systems.

10.2.3 Sound

It allows playing a sound, admitting files type WAV, MP3, MID, etc.

10.2.4 Windows services

Allows you to start, stop or restart Windows services. For security reasons, it requires the execution of Notify with elevated *Administrator* permissions. Otherwise, Windows will request acceptance from the current user, with the consequent problems in unattended systems.

10.2.5 Scheduled Tasks

Allows you to run, enable, or disable Windows scheduled tasks. It can be useful for running tasks that are already previously configured, or from other applications, such as triggering the generation of a Nexia Reports report.

10.2.6 Send SMS / email

Send an SMS or email message to a contact, a team, or any other recipient. The content of the message can be a free text, or a query configured in Notify.

10.2.7 InfluxDB: Writing

When triggered, data will be written to InfluxDB. You will need to configure:

- *Ret. Policy*: Name of the "retention policy". Optional.
- *Measurement*.
- *Time value*: Allows you to use a variable that indicates the timestamp for the data. Optional. By default, it uses the current time.
- *Time precision*: Enter the desired time precision. By default, one second.
- *Run only if data changes*: If this box is checked the insertion will only occur if there are any changes in the values of tags or fields, with respect to the last execution.
- *Tag set*: Enter here, one by one, the tags and values that will be associated to the data stored in InfluxDB.
- *Field set*: Enter here the field names and values to save.

The values can be a constant or a variable, alarm or event using substitution parameter syntax, and you can type them manually or select using the "Variable" or "Alarm / Event" buttons.

10.2.8 InfluxDB: Execute

Executes a generic statement written in InfluxQL. May be useful for deleting old data, for example.

10.3 Display

In visualization mode, in addition to consulting the configured actions, you can execute them manually. To do this, select it from the list, click on the "Execute selected command" button, or press F9. In the bottom bar of the window you can see the total number of configured actions, and the maximum available for your license.

10.4 Direct writing of a variable via message

In addition to configured commands, useful for writing multiple data simultaneously or for easy-to-remember commands, Notify allows you to write a single variable without the need to declare a specific command. To do this, send a message with the syntax *Name=Value*, where the name can be the full name or the alias of the variable, and the value must be valid for the specified variable type and range.



You can execute multiple actions or writes with a single message, separated by commas.

You can directly write a variable without having to configure a action, using the "Name=Value" syntax.

11 Notification / messaging systems

The main feature of the Notify application is the notification of alarms. For this, it has, according to its license, several messaging systems.

For a quick display of the current status, there is a status bar in the upper panel of the application. Clicking on the status icon gives access to a form that allows you to operate and check the status of each media separately. However, the configuration parameters have been centralized in the main *Settings* (see chapter 19-Setting).

11.1 Operation screen

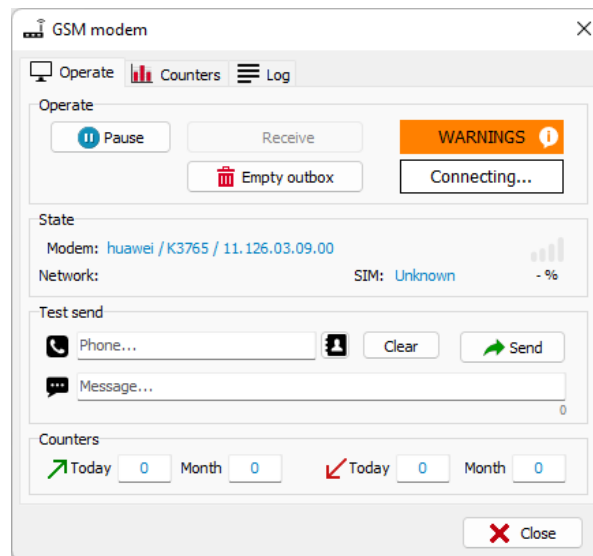


Illustration 24: GSM modem operation

Although there may be slight variations for each messaging system, this screen consists of several common tabs:

11.1.1 Operate

- **Operate**
 - *Pause / Resume*: Allows you to momentarily stop this messaging service.
 - *Receive*: Manually force the reception of new incoming messages.
 - *Empty outbox*: Cancel the messages pending to be sent, if any.
 - The current status of the service is shown, in text format and with a characteristic color.
- **State**: Displays data on the current status of the service. The information displayed depends on each system. For example, for a GSM modem the brand, model and version, the GSM network to which it is connected, the coverage level or the status of the SIM card will be displayed.
- **Test send**: Allows you to send a message manually, usually with the aim of verifying correct operation. The data requested depends on the type of message. You can select a contact from the phonebook or enter one manually. Pressing *Send* the message is added to the outbox, and will be sent as soon as possible.
- **Counters**: Displays the number of incoming or outgoing messages on the current day and month.

11.1.2 Counters

This tab allows you to check the number of messages sent or received in a period of time, in bar graph format. Select the direction, interval and message type, and move forward with the date buttons. By default, for daily interval, complete weeks (Mon-Sun) are displayed, and for the interval of months, they are displayed three by three.

11.1.3 Log

In this tab you can see real-time information about the execution of the service. If you experience a problem with sending or receiving messages, go to this tab for detailed information. This screen is limited to the last log messages

11.2 E-mail

The most basic messaging system available in Notify is email. For sending, the SMTP protocol is implemented, and for reception you can choose between POP3 or IMAP. SSL encryption and user authentication are also available.

Quick setup data has been added for the most widely used free mail servers: *Gmail*, *Yahoo*, *Outlook.com*, ..., but you can use any other provider that allows connection from external applications.

For more information about the configuration of sending and receiving emails, see the section 19.8.3-Setting-Email

11.3 GSM modem (SMS)

Notify has been designed to work with any modem connected to an RS-232 or USB serial port, which supports standard AT commands (Hayes command set⁶). The compatibility and correct operation have been tested with several commercial models (based on chipsets from Siemens, U-blox, Sierra Wireless or Cinterion). A local GSM modem allows you to both send and receive SMS.

Open to possible particularities of certain models, Notify provides the possibility of selecting between different controllers or drivers to adapt to possible variants in the syntax or availability of AT commands. Use the default driver, *Generic GSM Modem*, if the brand of your device does not appear among those available. If you experience any kind of problem, do not hesitate to contact Tecmatia or your supplier for help or information.

The coding systems of the GSM 03.38 standard (7, 8 and 16 bits) have been implemented, including the extended Spanish and Portuguese character tables of said standard. Keep in mind that the maximum length of an SMS depends on the selected encoding, and that the rate applied in case of using several SMS for a single message will depend on your operator. Notify allows you to select what to do in case a message does not fit in a single SMS.

⁶ The Hayes command set is a language developed by the Hayes Communications company that practically became an open standard of commands for configuring and parameterizing modems. Part of the Hayes command set was included by the ITU-T in the V.25ter protocol, currently V.250.

It is possible to set a limit of messages per day or per month, and even indicate on which day of the month the counter should be reset. There is associated a system alarm to indicate and notify if necessary that the established maximum is reached.



Notify works with any standard AT command compliant modem, connected to a serial (RS-232) or USB port.

For more information on configuring the GSM modem or the SMS sending system, see the section 19.8.2-Setting-GSM modem.

11.4 SMS online services (SMS gateways)

Another way to send SMS from Notify is using online services that provide an interconnection API. They are generically known as *SMS Gateways*. To use these services, you must have an Internet connection, and contract the service directly with each provider to later configure the access data provided by the provider in Notify.

Notify supports the following providers:

- *Twilio* (<https://www.twilio.com/>)
- *Vonage* (<https://www.vonage.com>)
- *Esendex* (<https://www.esendex.es/>)
- *Instasent* (<https://www.instasent.com/>)
- *BulkSMS* (<https://www.bulksms.com/>)
- *Mensatek* (<https://www.mensatek.com/>)
- *SMS publi* (<https://smspubli.com/>)
- *Altiria* (<https://www.altiria.com/>)
- *Clickatell* (<https://www.clickatell.com>)
- *Sendit* (<https://sendit.pt/>)
- *Lleida.net* (<https://www.lleida.net/>)
- *Sinch* (<https://www.sinch.com/>)
- *Textlocal* (<https://www.textlocal.com/>)
- *BulkGate* (<https://www.bulkgate.com/>)
- *GatewayAPI* (<https://gatewayapi.com/>)
- **Tecmatia** (<https://tecmatia.com/>). We also have our own service. Ask us about pricing and operation.

The implemented support only allows the sending of SMS, not the reception. For more information on configuring an SMS gateway or SMS sending system, see the section 19.8.3-Setting-SMS gateway (online services).



If you have a reliable Internet connection, Notify allows the sending of SMS with various platforms or online services, also called "SMS gateways".

11.5 External messages

It is possible to use Notify to send SMS or emails generated by other external applications. To interconnect applications a database is used, supporting practically any SQL database (Oracle, SQL Server, PostgreSQL, MySQL, Firebird, or ODBC). Notify can adapt to any existing system, since it does not require any specific table format or process flow. However, to obtain this flexibility, it is necessary to configure the SQL statements for the acquisition of messages to be sent and control of the result that adapt to your data structure.

11.6 Message history

The message history window allows you to consult all the incoming and outgoing messages of the application, with information on dates, recipients or senders, results, etc. Access this screen from the *Messages* section in the main menu.

Select the dates to consult picking from the quick list, or *Free* to specify any date range. Use the fields in the *Filter* panel to refine the search results, by message direction, type, service, contact, or specific text.

Message time	Destination / Source	Message	Service	Tries	Result	Date
17/11/2022 13:28:39		DEMO DESARROLLO - RE: Desde las 13:28 de hoy, y hasta desués, otra cos...	Email (prueb...	1	OK	17/11/2022 13:28:39
17/11/2022 13:28:33		Desde las 13:28 de hoy, y hasta desués, otra cos...	Email (prueb...		OK	17/11/2022 13:28:39
17/11/2022 13:24:57		Desde las 13:24 de hoy, y hasta desués, otra cos...	Email (prueb...		NOT ALLOWED	17/11/2022 13:25:17
17/11/2022 12:44:59		Desde las 14:03 de hoy	Email (prueb...		NOT ALLOWED	17/11/2022 12:45:13
17/11/2022 12:43:54		DEMO DESARROLLO - RE: Desde las 14:02 de hoy	Email (prueb...	1	OK	17/11/2022 12:43:55
17/11/2022 12:43:46		Desde las 14:02 de hoy	Email (prueb...		OK	17/11/2022 12:43:54
17/11/2022 12:42:52		DEMO DESARROLLO - RE: Desde las 14:01 de hoy	Email (prueb...	1	OK	17/11/2022 12:43:04
17/11/2022 12:42:40		Desde las 14:01 de hoy	Email (prueb...		OK	17/11/2022 12:42:52
17/11/2022 12:29:58		DEMO DESARROLLO - RE: Desde las 12:30 de hoy	Email (prueb...	1	OK	17/11/2022 12:29:59
17/11/2022 12:29:43		Desde las 12:30 de hoy	Email (prueb...		OK	17/11/2022 12:29:58

Illustration 25: Message history

The list shows following data:

- **Message direction:** Icon that indicates if it is a sent or received message.
- **Type of message:** Icon that represents the type of message (SMS, Email ...).
- **Message time:** Creation date of the message.
- **Destination / Source:** You will see the destination (sent) or source (received) phone number or email. If the contact is in the phonebook, the name will also be displayed.
- **Message:** Message content. For email, only the subject field is displayed.

- **Result:** Descriptive text of the result of the sending or receiving operation. Can be:
 - *OK*: The message has been sent or received correctly.
 - *ERROR*: There was an error sending or receiving the message. See the log files for more details.
 - *CANCELLED*: Sending was cancelled, because the message service was stopped, or the outbox was manually emptied.
 - *NO LICENSE*: You do not have a license for the messaging system.
 - *LIMIT*: The message was not sent because the configured maximum or monthly limit was reached.
 - *TIMEOUT*: The sending has been cancelled because the message could not be sent in the maximum period of time configured.
 - *DISABLED*: The message has not been sent because the sending system is disabled.
 - *NOT SERVED*: For incoming messages, indicates that the message has not produced a response from Notify because there is no query, variable or script associated with the received text.
 - *NOT ALLOWED*: For incoming messages, indicates that the sender does not have permission to execute the query, write or action.
- **Date:** Time when the message was sent or received.

You can export the query results to a file in CSV format.

11.7 Periodic status report

It is possible to schedule the periodic sending of a status message, in order to remotely monitor the application and the status of all the services, as well as report alarms active or other system information. If you do not receive the message at the configured time, it will be a sign that the computer is turned off, application is not running, or something is not working correctly.

In addition to the general configuration, where you can enable the types of messages or configure the content and the schedule, you must assign the system event “*Notify - State report*” to the contacts you want to receive the message.

12 Contacts

To notify alarms, or to assign write or query permissions, it is necessary to define one or more contacts. These can be classified into groups or teams and added in notification policies. Access the *Contacts* section from the main menu to view or edit them.

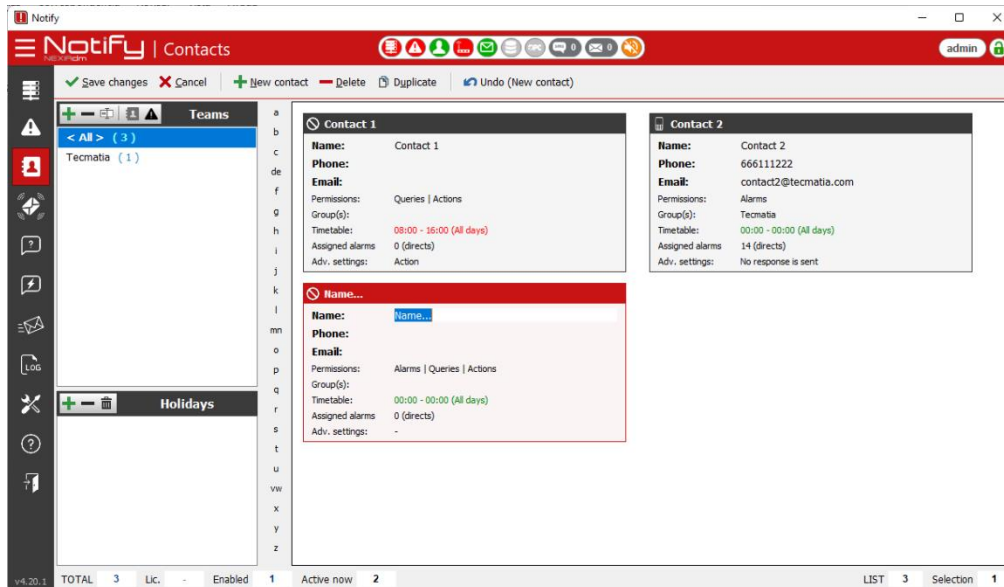


Illustration 26: Contacts.

12.1 Display

This section shows the *"Teams"* and *"Holidays"* sections on the left side and the list of contacts is shown in the main area of the screen. In the *Teams* section the created contact groups or teams are displayed. When selecting a team, only those assigned to it are displayed in the contact list. To view all the contacts, select *"< All >"* on the team selector.

The *"Holidays"* section contains the special days that are declared as holidays.

The contacts are displayed in a card list format, where each one represents a declared contact and displays the relevant information about it. If the number of declared contacts is large, you may find the index sidebar that gives quick access starting from the initial of the name useful. You can also use the keyboard, since pressing a key will locate the 1st tab whose name begins with that letter.

The card shows in a legible format the timetable configured for the contact, as well as the number of assigned alarms. For more information, double click on the field and a window with more detail will open.

In the bottom bar of the window, you can see the total number of contacts declared, and the maximum available for your license. You can also see how many are enabled for alarm notification and how many are in active hours right now.

12.2 Editing

12.2.1 Teams

Pressing "*Enable editing*" of the header goes to edit mode. In this mode, by accessing the popup menu (right mouse button) or the button panel of the team's section, you can:

- **Add / Delete / Change name:** Add and delete teams or change the name.
- **Assign contacts:** It allows associating the contacts that belong to the team by selecting them from a list.
- **Assigned alarms:** Allows you to select the alarms directly assigned to the team, that is, all members of the group will be notified.

12.2.2 Contacts

Each contact card has the following fields:

- **Name**
- **Phone:** For sending SMS. You can enter the phone number in the format that is most convenient for you: With or without international prefix, with or without spaces ... Notify will correctly format the number when sending an SMS.
- **Email:** Please enter a unique valid email address.
- **Permissions:** Select what permissions the contact will have:
 - *Queries:* Indicate if the contact has permission to make queries by SMS / Email.
 - *Actions:* Indicate if the contact has permission to perform writings or actions by SMS / Email.
 - *Alarms:* Indicate whether alarms will be reported to this contact.
- **Team(s):** Indicate which groups the contact belongs to. You can assign none, one or more.
- **Timetable:** Periods in which the contact is active for alarm notification. It does not affect the permission for queries or actions.
- **Assigned alarms:** List of alarms to be notified to this contact. Note than in query mode, all assigned alarms appear, either directly, assigned to team, or assigned to a notification policy. In edit mode, only direct assignments appear.
- **Advanced settings:**
 - *Received messages:*
 - **Split messages:** By default, the received messages are segmented, separated by ',' and ';', and are processed individually as if they were several messages. This can be useful for doing multiple queries or actions with a single message. However, this may not be of interest in certain contexts, such as automatically generated messages that need to be processed together. Uncheck this box if it is the case.
 - **Execute action...** For every message received from this contact. Further, marking "...and do not process", it will only execute the action and the message will not be processed (to see if it is query, acknowledge, etc.)
 - **Forward/... and do not process:** Forwards the received message to an contact list. If it is marked "and do not process", Notify will not process the content of the message to interpret queries or actions.

- **Ignore / Just accept pattern:** Allows a filtering of received messages, using a fixed text, an MS-DOS pattern (with * and ? wildcards) or a regular expression (RegEx).
- **Do not send an OK / ERROR response:** Check these options if you want the messages from this contact not to be answered, in any case, informing whether they have been processed correctly (OK) or not (ERROR).
- *Outgoing messages:*
 - **Add alarm identifier to facilitate acknowledge "... (#N)":** By default, the alarm's ID number is added to the activation messages that require acknowledgment, facilitating the response. If you wish not to add this number so as not to alter the original message, uncheck this.
 - **Execute action:** For every message sent to this contact.
- *Notify login:*

Enables the contact to log in to Notify, and at what user level. In this case the username can be the name of the contact or the email. you must enter a password if access is enabled.
- *Missed call:*

It allows you to enable or not, and configure what action to associate with the receipt of a missed call by this contact. It can be the acknowledge of pending alarms, a query (user or system), or execute an action.

Enabling the edit mode and accessing the popup menu (right mouse button) or via the shortcut on the top button, you will have the following options:

- **New contact:** Add a new contact. Add a card to the list and jump directly to the *Name* field for you to enter the name of the new contact.
- **Delete:** Remove the selected contact.
- **Copy / Paste / Duplicate:** It allows you to quickly duplicate one or more contacts, which you can later modify.
- **Paste special:** After copying a contact, this option will allow you to apply the assigned timetable or alarm settings to another contact you select.
- **Edit fields:** In the fields of the contact card, press directly on it to be able to modify its value. Note that the title of the tab is not directly editable, it corresponds to the *Name* field, which is the one that can be edited.
- **Permissions / Assign to teams... / Assigned alarms...:** Access these settings through the context menu, or by double-clicking on the corresponding field on the tab.
- **Timetable:** Opens the configuration window for the contact's active timetable. In this window you can add or modify the time periods in which the contact is active for alarm notification. A start and end time is defined for each period, as well as the days of the week, months, weeks, or holidays in which it applies. You can also access by double clicking on the *Timetable* field in the contact card.
- **Holidays:** If needed, Notify allows you to enter a list of special days or holidays and indicate if the contact is also active on those holidays or not. To configure the list of holidays, use the "Holidays" section found in the bottom left of the screen. To indicate if a contact is active on these days, check or uncheck the checkbox "Holidays" in the configuration of an individual timetable period.

- **Undo:** You can undo the last editing operation.



Notify allows a very flexible configuration of active contact hours. You can indicate times, days of the week, distinguish odd and even weeks, months, and even holidays.

The image shows a software dialog box titled "Edit Timetable" with a close button (X) in the top right corner. The dialog is organized into several sections: "Days" with checkboxes for L, M, M, J, V, S, D, a "Holidays" dropdown menu set to "Indifferent", and an "All" checkbox; "Hours" with "From" and "To" time pickers both set to "00:00" and a "24h" button; "Months" with checkboxes for numbers 1 through 12 and an "All" checkbox; and "Weeks" with radio buttons for "All", "Odd", and "Even". At the bottom, there are "OK" and "Cancel" buttons.

Illustration 27: Timetable settings

13 Historical data



The setting for data log at the variable level (deadband, times, etc.), affects both the internal data log and the data log in an external database.

13.1 Internal data log

Notify has as standard the function of storing the values of variables and alarms in an internal database. Your license may limit the maximum number of days for this data. The internal database is in a proprietary format binary files, separated by days and optimized for quick access to obtain efficient graphs and reports.

The alarm data log is always enabled, while the variable data log can be disabled or limited if not needed.



The history in the internal database is the one that allows Notify to perform the graphs and queries of historical data, even if its license does not enable the registration in an external database.

13.2 Data log in external database (SQL)

Optionally, if your license enables it, Notify can store historical data in an external RDBMS-type database. By default, it provides native support for Oracle, SQL Server, MySQL, PostgreSQL, SQLite, or Interbase/Firebird databases, as well as indirect support for any other that has an ODBC driver⁷.

The configuration of this feature is intended to be generic enough to allow it to adapt to any specific data structure:

- The recording of variables or alarms can be enabled independently.
- The databases of variables and alarms can be the same or different.
- You can configure the SQL to be executed to record the value of a variable, the activation or deactivation of an alarm. A minimum knowledge of SQL language is required. Do not hesitate to ask Tecmatia for any help or information you need.
- You can configure a periodically executed SQL.
- You can enter Null marks at the beginning or end of the data.
- You can force data logging at the start of each day, to ensure that there is always some value in the day.

In the section **¡Error! No se encuentra el origen de la referencia.**-Setting-**¡Error! No se encuentra el origen de la referencia.** you will find more information on configuring this feature.



Notify allows you to store historical data in the most used RDBMS (relational) databases (Oracle, Microsoft SQL Server, MySQL, PostgreSQL, Interbase / Firebird), or any other that has an ODBC driver. It can also store data in InfluxDB, a powerful and versatile TSDB (Time Series Database) database.

⁷ODBC: Open Database Connectivity. Database access standard available on Microsoft systems.

13.3 Data log in external database (InfluxDB)

In addition to SQL, Notify also allows the ingestion of data in an external database of type TSDB (Time Series Database). Specifically, it supports *InfluxDB*⁸ (version 1.x and 2.x). It is a database available in *open source* version, with excellent performance and powerful functionalities, strongly recommended by Tecmatia, in joint use with *Grafana*⁹, an attractive and versatile web application for graphing and monitoring data. To produce reports in table format oriented to printing, Tecmatia has a powerful tool of its own development, *Nexia Reports*, since for this functionality no satisfactory existing solutions have been found on the market.

We invite you to know these tools. You will be surprised by the great possibilities they offer to your project and the ease of use and start-up. Consult Tecmatia for more information.



Illustration 28: Historian in InfluxDB + Grafana

⁸ InfluxDB is a product of InfluxData. More information at <https://www.influxdata.com/>

⁹ Grafana is a product of Grafana Labs. More information at <https://grafana.com/>

13.4 Internal graphs

From the data stored in the internal database, Notify provides you with a graphing tool that allows you to analyse the evolution of any variable or alarm. You will access it from the *Variables* or *Alarms* section by selecting the one you want and pressing the graph button at the top panel, in the popup menu, or pressing the F10 key.

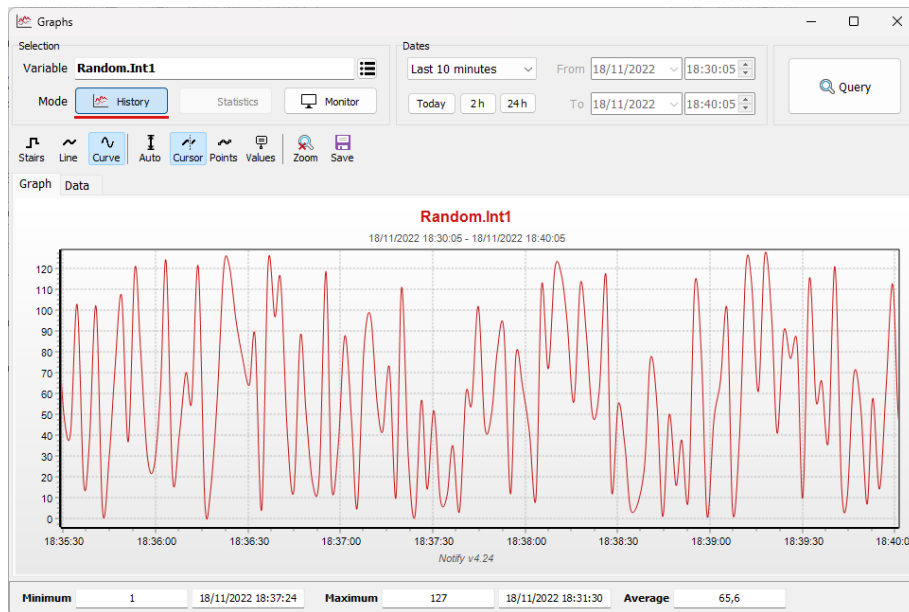


Illustration 29: Graph for one variable

13.4.1 Selection panel

- **Variable / Alarm:** Allows you to select another variable or alarm. One mode or another is determined by whether the graph is called from the *Variables* or *Alarms* section, and cannot be exchanged here.
- **Mode:** Graph display mode.
 - **History:** Allows you to consult historical data between two specific dates.
 - **Statistics:** Available for alarms and for digital or counter variables. Displays a bar graph with the statistical data by intervals between two selected dates.
 - **Monitor:** In this mode, the graph scrolls and updates automatically, showing the evolution of the variable or alarm in the last period of time indicated. When selecting this mode, you have a field that allows you to select the number of minutes to display, the option to pause or resume automatic advance, and a field that displays the current value of the variable or alarm.
- **Date selection:** You have quick access to typical date ranges (today, yesterday, last hour, last month, etc.). If you want to indicate any two dates, select the Free option, which enables the From and To fields so that you can enter the desired day and time.
- **Interval selection:** For the *Statistics* mode, there is a field that allows you to select whether you want to display the data by hour, by day or by month.
- **Query:** Press the button to update the content of the graph when changing the selection parameters.

13.4.2 Graph and data panel

In the *Graph* tab, you have these options:

- **Stairs:** The graph is displayed in a “flat” linear format, such that the line remains constant until the next value change. It is the only possible mode for digital variables and alarms.
- **Line:** In this mode, the graph “joins” the different data samples in a straight line, smoothing the value transitions, and if the real variable of measurement evolves progressively.
- **Curve:** In this mode the graph “joins” the samples into a simulated curved line, further smoothing the display. Note that not all data types allow the simulation of a curve, which can result in an inconsistent display with certain variables. Use the mode that best suits the actual meaning of the variable.
- **Auto:** Automatically adjusts the vertical scale, adapting to the magnitude of the variable. If it is not checked, the scale will be determined by the range indicated in the variable configuration.
- **Points:** Show or hide the points on the graph that represent the logged samples.
- **Values:** Shows or hides a label associated with each sample with its value. Useful for graphs with few value changes.
- **Save:** Saves the graph to a file, in JPEG or BMP format, or data points in CSV or Excel format.

The title of the graph is always formed by the name of the variable or alarm, including the full path if necessary (server, group, folder), and the dates of the time interval to which it refers.

The scale of the Y axis adapts the type of variable and the content. By default, for non-digital variables, the values configured in *Range - Minimum - Maximum* of the variable are taken.

In addition to viewing the graph, you can query the *Data* tab, where you will see the evolution of the value in table mode, in time-value format. This table allows you to sort or export the data to a CSV file¹⁰.

13.4.3 Statistics panel

In the bottom side, a panel with the statistical data of the variable or alarm for the selected date period is shown as a summary. The displayed values depend on the type of variable:

- **Digital variable:** The time at '0', the time at '1', and the number of activations (value changes from '0' to '1') are displayed.
- **Analog variable:** The minimum, maximum and average value in the period are displayed.
- **Counter variable:** The minimum, maximum and count value in the period are displayed. The count is calculated as the sum of all positive increments of the value.
- **Alarm:** The active time, the inactive time, and the number of activations in the period are displayed.

¹⁰CSV: Comma Separated Values. Plain text file type where each line represents a row of data, and each value is separated by a comma (,). The standard says that text values containing spaces or commas must be enclosed in double quotes (“”). It is also possible that the data is separated by a semicolon (;) instead of the comma, this being the best format to work with Excel.

14 OPC-DA server

Notify integrates an internal OPC-DA server, which allows interacting and sharing information with external applications, such as a SCADA. Data like the current value of variables, alarm status, contact information, and some configuration and status parameters are available.

This is a powerful feature, since it enables Notify to act as a protocol gateway (Modbus to OPC-DA, for example), or to serve internal or transformed variables. It also possible activating or deactivating Notify functions, like enabling alarms or contacts, change email or phone number, etc. This can be useful to create a unified user interface in the SCADA integrating Notify.



Notify has an internal OPC-DA server, which allows the value of variables, alarm status, application status, etc. to be shared with external applications. It also allows writing variables, configuration parameters or actions from the external application.

Some of the OPC variables available:

- **Alarms:** Current status as boolean (True / False).
- **Application:** Application data and parameters.
 - *AlarmsMute:* Indicates if there are silenced variables. Allows writing to silence all alarms (True), or cancel silence (False).
 - *AlarmsSend:* If the alarm notification is enabled. Writeable.
 - *CountAlmTotal:* Number of declared alarms.
 - *CountAlmEnabled:* Number of alarms enabled.
 - *CountAlmActiveAlarms, CountAlmActiveEvent, CountAlmActiveTotal:* Number of active alarms except events, only events, or all included, respectively.
 - *CountAlmMute:* Number of silenced alarms.
 - *CountAlmUnack:* No. of unacknowledged alarms.
 - *CountContactsTotal:* Total number of contacts.
 - *CountContactsActive:* Number of active contacts now (within active hours).
 - *CountContactsEnabled:* Number of contacts enabled to send alarms.
 - *CountVarTotal:* Total number of variables.
 - *CountVarEnabled:* Number of enabled variables.
 - *CountVarBad:* No. of variables with *bad quality* value.
 - *CountVarOk:* Number of variables with good value.
 - *DataClientsOK, DataloggerOK, EmailOK, SmsOK:* Indicates whether the data client, data logger service, emails or SMS modules are working properly.
 - *EnableEmail:* Writeable, to enable or disable sending and receiving emails.
 - *EnableSMS:* Writeable, to enable or disable sending and receiving SMS.
 - *EnableStateMsg:* Writeable, to enable or disable state report message.
 - *EnableSound:* Writeable, to enable or disable sound.

- *ProjectName*: Project's name.
- *UserName*: Name of the authenticated user.
- **Contacts**: Data of declared contacts. Allows writing to modify externally.
 - *Enabled*: If alarm notification is enabled for the contact.
 - *Name*: Name.
 - *Phone*: Telephone number.
 - *Email*: Email.
- **DataClients**: Data client information:
 - *Name*: Name.
 - *Type*: Client or protocol type.
 - *Enabled*: Enabled.
 - *Active*: Active now. Normally it is equivalent to *Enabled*, except manual deactivation.
 - *Connected*: On-line now.
 - *Ok*: Working properly.
 - *StatusText*: Current state, as descriptive text.
 - *CountVarTotal*, *CountVarEnabled*, *CountVarOk*, *CountVarError*, *CountVarBad*: Total number of variables, enabled, with correct reading, with reading error, or with bad quality value, respectively.
- **Senders**: Information on the notification / messaging systems available:
 - *Enabled*: Enabled.
 - *Connected*: Connected.
 - *Paused*: Pause.
 - *Status*: Current state as number. (0=Initialize, 1=Disabled, 2=Disconnected, 3=Connecting, 4=Connected, 5=Sending, 6=Retrieving, 7=Disconnecting, 8=Paused)
 - *StatusText*: Current state as text.
 - *OperatingStatus*: Operational state as number. (0=Stopped, 1=Unsupported, 2=Disabled, 3=Paused, 4=Warning, 5=Error, 6=OK)
 - *OperatingStatusText*: Operational state as text.
 - *LastSendOK*, *LastReceiveOK*: Date of the last message sent or received.
 - *MsgCounterDayIN*, *MsgCounterDayOUT*: Daily counter of received and sent messages.
 - *MsgCounterMonthIN*, *MsgCounterMonthOUT*: Monthly counter of received and sent messages.
- **Variables**: Status and current value of all variables.

15 API REST

Since release 4.18, Notify has a REST API¹¹ that is useful to be able to perform actions from external applications, for example from Grafana, or interact between different instances of Notify, reading or writing data. Authentication is done using *API keys*, which the client must know in order to access the server. To increase security, allow you to use HTTPS.

See section 19.4.3 for more details about configuration.

15.1 Base URL

http(s)://server:port/notify/api/...

15.2 API key

They are generated from the configuration window in Notify. They can be read-only, or read/write. Several keys can be generated, for different clients or different uses. To revoke the access, you only need to remove the key from the list in the settings.

The *API key* is required for all API calls. It can be indicated as the first parameter in the URL, just after the *endpoint*, or as an HTTP header with the *x-api-key*.

15.3 HTTP results

Complying with the REST standard, the result of each request is returned as a standardized HTTP status code, these being the possible ones:

- 200: OK.
- 400: Error in request. It can be an *endpoint* that does not exist or an error in the parameters.
- 401: Unauthorized. *Api key* missing or invalid.
- 403: Not allowed. The operation you intend to perform does not have permission. For example, a write with a *read-only api key* .
- 404: Not found. The element or resource does not exist (for example, when typing a variable).
- 500: Internal server error. See the server-side log for more details.

15.4 Endpoints

Read functions are performed with GET and writes with POST. In this manual where otherwise noted refers to the GET command.

15.4.1 Alarms

(Aliases allowed on the endpoint base: "alert", "alerts", "alm", "alarm", "alarms")

- **.../alert/list/[api-key]/[filter]**

Alarm list.

¹¹ REST API: Application programming interface (API) that conforms to the limits of the REST architecture: Client-server architecture; requests over HTTP; stateless communication; uniform interface between elements; information in JSON or plain text.

[filter]: Optional. One or more filter conditions, separated by commas:

- id=x
- enabled, disabled, active, inactive, ack, unack, system, user, mute, hidden
- prio=x, minprio=x, maxprio=x ('Event', 'Low', 'Normal', 'High', 'Critical')
- folder=x
- contact|contactid=x (assigned to a contact with this id)
- name|pattern|namepattern=x (filter by name, with '*' and/or '?')
- regex|nameregex (filter by name, with regular expression)

- **.../alert/read/[api-key]/[ids]/[filter]**

Alarm status reading.

[ids]: One or more identifiers, separated by commas. The id can be the internal numeric identifier or a text with the alias or name.

- **.../alert/status/[api-key]/[ids]**

Alarm status reading. Reduced version with less information in the response.

- **.../alert/folders/[api-key]**

List of alarm folders.

15.4.2 Variables

(Aliases allowed on the endpoint base: "var", "vars", "variable", "variables")

- **[POST] .../var/write/[api-key]/id/value**

Writing a variable. The id can be the internal numeric identifier or a text with the alias or name.

- **[POST] .../var/write/[api-key]/id1=value1,id2=value2...**

Writing one or more variables.

- **.../var/list/[api-key]/[filter]**

Variable list, in flat list.

[filter]: Optional. One or more filter conditions, separated by commas:

- id=x
- enabled, disabled, writeable, good, bad, error, ok
- vk|kind=x (filter by variable kind: 'B'=Digital, 'A'=Analogical, 'C'=Counter, 'H'=Hex, 'D'=Date, 'T'=Text, 'N'=Number) (supports more formats)
- units=x
- folder=x
- name|pattern|namepattern=x (filter by name, with '*' and/or '?')
- regex|nameregex (filter by name, with regular expression)

- **.../var/listtree/[api-key]/[filter]**

List of variables, with result in list in JSON tree mode (data client > group > variable).

- **.../var/read/[api-key]/[ids]/[filter]**

Reading variables, with current value.

[ids]: One or more identifiers, separated by commas. Allows numeric internal id, or alias or name text.

- `.../var/value/[api-key]/[ids]/[filter]`
Reading variables, with current value. Reduced version with less information in the response.
- `.../var/folders/[api-key]`
List of variable folders.

15.4.3 Internos

- `.../internal/read/[api-key]/ids/[filter-text]/[filter-folder]/[filter-tristate]/[filter-prio]`

For internal use between different instances of Notify. Allows the reading of internal data, variables or alarms.

- `.../internal/ping/[api-key]`
Check connection to API.

15.4.4 Actions

(Aliases allowed on the endpoint base: "action", "actions")

- **[POST]** `.../action/execute/[api-key]/id`
Execute an action.

15.4.5 Messages

(Aliases allowed on the endpoint base: "msg", "message", "messages")

- `.../msg/list/[api-key]/fromDateTime/toDateTime/[type]/[direction]`

List of messages sent and/or received.

[fromDateTime] , [toDateTime]: Start and end date. Supports different formats:

- yyyyymmdd
- yyyyymmddhhnn
- yyyyymmddhhnnss
- Local format (avoid separator '/')
- ISO8601

[type]: Message type (optional) → · sms · email · all

[direction]: → · in: Messages received. · out: Messages sent.

16 Web Access

Notify does not have its own web application. However, web access to variables and alarms is possible with Grafana and InfluxDB. To do this, it is necessary to have the external data logger available in the license, and enable “standard Notify data logger” for InfluxDB in main *Settings*.

You will find the current value and information of all the variables in the `_Notify_VarSummary` measurement, which you can display as you need by creating your own dashboards.

For alarms, a typical installation displays the information on three Grafana dashboards:

- **Notify - Summary:** Shows the current status of the alarms, being able to filter by groups or status.

Notify > Alarmas - Sumario

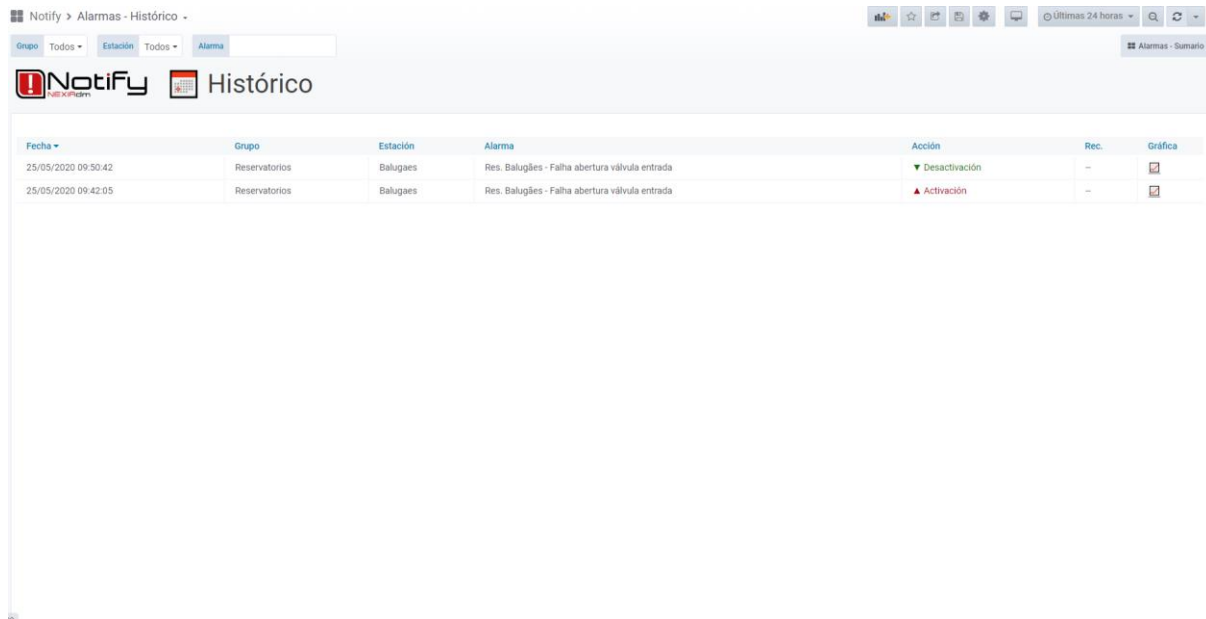
Grupo: Todos | Estación: Todos | Estado: Todos | Alarma: []

TOTAL: 433 | Deshabilitadas: 40 | Activas: 7 | No reconocidas: -- | Silenciadas: --

Grupo	Estación	Alarma	Estado	Fecha activación	Rec.	Gráfica
EE	Medros	EE Medros - Corte geral	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Falha válvula entrada	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Caudal saída anomalia	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Caudal saída baixo	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Caudal saída alto	▲ Activa	17/03/2020 17:32:43	--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Cloro anomalia	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Cloro baixo	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Cloro alto	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Nivel reservatorio 1 anomalia	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Nivel baixo reservatorio 1	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Nivel alto reservatorio 1	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Falho analisador cloro	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Baixo nivel cloro	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Intrusão	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Baixa bateria	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Balugas	Res. Balugães - Falho tensão rede	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Faial	Res. Faial - Bóia máximo	✓ Inactiva		--	<input type="checkbox"/>
Reservatorios	Faial	Res. Faial - Bóia mínimo	✓ Inactiva		--	<input type="checkbox"/>

Illustration 30: Grafana - Notify Dashboard Summary

- **Notify - Historical:** Shows the activation, deactivation, and acknowledgment events of the alarms, in the given range of dates.



Notify > Alarmas - Histórico

Últimas 24 horas

Grupo: Todos Estación: Todos Alarma:

Notify Histórico

Fecha	Grupo	Estación	Alarma	Acción	Rec.	Gráfica
25/05/2020 09:50:42	Reservorios	Balugaes	Res. Balugães - Falha abertura válvula entrada	Desactivación	-	
25/05/2020 09:42:05	Reservorios	Balugaes	Res. Balugães - Falha abertura válvula entrada	Activación	-	

Illustration 31: Grafana - Historical Notify Dashboard

- **Notify - Graph:** This panel can be accessed from the previous two, by clicking on the symbol of the graph of one alarm. It shows a simple graph with the evolution of the alarm over time where 0 represents an inactive alarm and 1 an active alarm.



Illustration 32: Grafana - Graphical Notify Dashboard

17 Substitution parameters

A substitution parameter is a way of referencing dynamic data in a configuration text box, in a message template, etc. Notify allows you to use parameters in some places. These are enclosed in curly brackets, with the generic syntax `{group.field@modifiers}`, where “group” and “modifiers” are optional. So, `{field}`, `{field@modifiers}` or `{group.field}` are also valid syntaxes. It is not case sensitive.

Although the syntax is common to several Tecmatia applications, the parameters available in Notify are detailed below.

17.1 Generic data

- **now**: Current date.
- **utc**: Current date in UTC.

17.2 PC data

- **pc.name**: Computer or PC name (default¹²).
- **pc.ip**: Local IP of the PC.
- **pc.publicip**: Public IP of the PC.
- **pc.hwid**: HW identifier of the PC (valid for Tecmatia licenses).
- **pc.internet**: Internet available (Boolean).
- **pc.os**: Name of the operating system.
- **pc.winuser**: Windows session user name.
- **pc.winpk**: Windows Product Key.
- **pc.ram**: Total memory (in text).
- **pc.processor**: Processor (name in text).

17.3 Application data

- **app.name, app.title**: Name of the application (Notify).
- **app.version**: Application version.
- **app.edition**: Name of the edition of Notify (Lite, Standard, ...).
- **app.user, app.username**: Current user.
- **app.userlevel**: Current user level, as number (0..5).
- **app.userrol**: Current user level, as text.
- **app.projectname**: Name of the project or facility.
- **app.sepcsv**: Separator configured for CSV.
- **app.path**: Data path.
- **app.configpath**: Configuration files path.
- **app.templatepath**: Message template path.
- **app.logpath**: Log file path.
- **app.statepath**: Current state file path.
- **app.temppath**: Path of temporary files of the application.
- **app.backuppath**: Backup path.
- **app.starttime**: Application start date.
- **app.ram**: Application memory usage.

¹² Default field: This annotation means that the substitution parameter will return the value of that field if field name is missing. For example, `{computer.name}` and `{computer}` are equivalent.

17.4 Variables Data

The syntax to refer to a variable can be:

- **{field}**: In the context of a variable, it refers to the variable itself.
- **{v.field}**: In the context of an object that can contain a variable, it refers to that variable. For example, in an alarm, in a write action, etc.
- **{vNNN.field}**: Reference to a specific variable, given by the internal id "NNN" (number).

If the field name is omitted (writing {v} or {vNNN}), the default field is "value".

Available fields:

- **id**: Internal id.
- **name**: Name.
- **alias**: Aliases or short text.
- **enabled**: Enabled.
- **value**: Value of the variable, in format according to the type of variable.
- **influx**: Value in valid format for the InfluxDB line protocol, according to the type of the variable. Dates are treated as a Unix Timestamp integer in UTC.
- **active**: Variable is "active", that is, has a non-zero value.
- **astext**: Value formatted as text.
- **asboolean**: Value as Boolean (True/False). Equivalent to "active".
- **as01**: Value as Boolean "1" / "0".
- **asfloat**: Value as Float or Real.
- **asint**: Value as an integer.
- **asdate**: Value as date.
- **path**: Folder (full path from root).
- **fullpath**: Folder + Variable name.
- **folder**: Only last folder in the path.
- **root**: Root folder (only first in the path).
- **server**: Name of the data server.
- **group**: Name of the reading group within the server.
- **units**: Units.
- **item, address**: Item name or reading address.
- **idext, extid**: Id for external registration.
- **time**: Date of current value.

Information about the variable list:

- **vars.count**: Total variables.
- **vars.enabled**: Total of enabled variables.
- **vars.ok**: Total variables OK.
- **vars.error**: Total variables with reading error.
- **vars.bad**: Total of variables with bad quality.

17.5 Alarm / Event Data

The syntax to refer to an alarm or event can be:

- **{field}**: In the context of an alarm, it refers to the alarm itself.
- **{a.field}**: In the context of an object that can contain an alarm, it refers to that alarm. For example, in a message template.

- **{aNNN.field}**: Reference to a specific alarm, given by the internal id “NNN” (number).

If the field name is omitted (writing {a} or {aNNN}), the default value is “active”.

Available fields:

- **id**: Internal id.
- **name**: Name.
- **alias**: Alias.
- **enabled**: Enabled.
- **value, active**: Active current state (default field).
- **as01**: Current state active field "1" / "0".
- **state, status**: Current state (descriptive text).
- **influx, influxb**: Active state in Boolean format valid for InfluxDB line protocol.
- **influxi**: Numeric status in Integer format valid for InfluxDB line protocol.
- **path**: Folder (full path from root).
- **fullpath**: Folder (full path from root) and alarm name.
- **folder**: Folder (only last in the path).
- **root**: Root folder (first in path).
- **condition**: Alarm condition (descriptive text).
- **alarmvalue**: Alarm value.
- **priority**: Priority, in text.
- **stable**: Stabilization time for active (text, time literal).
- **idext, extid**: Id for external registration.
- **atime**: Activation date.
- **dtime**: Deactivation date.
- **rtime**: Acknowledgement date.

Information about the alarm list:

- **alerts.count**: Total alarms.
- **alerts.active**: Total active alarms.
- **alerts.events**: Total active events.
- **alerts.enabled**: Total alarms enabled.
- **alerts.timeenabled**: Total alarms enabled now.

17.6 Contacts Data

The syntax to refer to a contact can be:

- **{c.field}**: In the context of an object that can contain a contact, it refers to that contact. For example, in a message template.
- **{cNNN.field}**: Reference to a specific contact, given by the internal id “NNN” (number).

If the field name is omitted (writing {c} or {cNNN}), the default value is “name”.

Available fields:

- **id**: Internal id.
- **name**: Name (default field).
- **phone**: Telephone.
- **email**: E-mail.
- **active**: Active now. Indicates if available at that moment to receive notifications.

The following information is available from the contact list:

- **contacts.count**: Total contacts.
- **contacts.enabled**: Total enabled contacts.
- **contacts.active**: Total active contacts now.

17.7 Message Data

In certain contexts, such as the actions associated with the receipt of a message, it is possible to refer to its content, using the syntax:

- **{msg.field}**: In the context of an object that can contain a message, it refers to that message.

Available fields:

- **text**: Message text. (Default).
- **time, datetime**: Date of the message (complete).
- **date**: Message date (day only).
- **sender, source**: Sender.
- **recipient, addressee**: Addressee.
- **ok**: Received / Sent successfully (True / False).
- **result**: Result of sending or receiving (text).
- **type**: Type of message ('SMS', 'Email', ...).
- **priority**: Priority ('Lowest', 'Low', 'Normal', 'High', 'Highest').

17.8 Modifiers

The value of a specific field can be transformed using one or more “modifiers”. The possibility of using some modifiers or others will depend on the type of data on which it is applied, so special care must be taken to avoid an anomalous or unknown result ("?").

Syntax: **{group.campo@modif1@modif2...}**

Possible modifiers are:

- **(N)**: Substring. If $N > 0$, the first N characters. If $N < 0$, the last N characters.
- **(N, M)**: Substring, gets M characters starting from position N .
- **upper**: Upper case.
- **lower**: Lower case.
- **quoted**: Text enclosed in single quotes.
- **dquoted**: Text enclosed in double quotes.
- **length**: Length of the text.
- **split(sep,n)**: Segments or splits the text delimited by the characters entered in "sep", and returns segment number "n".
- **before(sep)**: Returns the text before the "sep" separator.
- **after(sep)**: Returns the text after the "sep" separator.
- **trim**: Remove spaces at the beginning and end.
- **nosp**: Remove all spaces.
- **sp_**: Replace spaces with '_ '.
- **cap**: First uppercase letter in all words, lowercase rest.
- **cap1**: First uppercase letter in the first word, lowercase rest.
- **[]**: Returns the value of the index if it is an array with format [N] (first occurrence).
- **#** or **#1**: First number found in the text.
- **#L**: Last number found in the text.
- **add(N)**: Adds N to a numeric value (float allowed).

- **b**: Value as Boolean "True" / "False".
- **b01**: Value as Boolean "1" / "0".
- **bt, byn, bsn**: Value as Boolean "T" / "F", "Y" / "N" or "S" / "N", respectively.
- **b(VT, VF)**: Value as Boolean, indicating specific texts for true (VT) and false (VF).
- **i**: Value as an integer. If the source data is a date, in Unix Time.
- **0x**: Value in hexadecimal format.
- **f**: Value in float, using local decimal separator.
- **. or f.**: Value in float, using decimal separator "."
- **, or f,**: Value in float, using decimal separator ","
- **f(format)**: Value in float, with specific format. Indicate valid format for Delphi *FormatFloat* function.
- **influxF**: Value as float for InfluxDB line protocol (decimal point is ".").
- **influxB**: Value as a Boolean for InfluxDB line protocol (T / F).
- **influxI**: Value as integer for InfluxDB line protocol ("i" is added at the end).
- **influxS**: String value for InfluxDB line protocol (quotes).
- **utc**: Value interpreted as UTC date, and expressed as an integer Unix Time in seconds.
- **utcms**: Value interpreted as UTC date, and expressed as an integer Unix Time in milliseconds.
- **unix**: Value interpreted as a local date, and expressed as a Unix Time integer in seconds.
- **unixms**: Value interpreted as a local date, and expressed as an integer Unix Time in milliseconds.
- **date**: Value interpreted as date, and formatted as only date.
- **time**: Value interpreted as date, and formatted as only time.
- **t**: Value interpreted as date, and formatted with date and time.
- **ts**: Value interpreted as date, and formatted as 'yyyymmddhhnss'.
- **t(format)**: Value interpreted as date, and specific format, valid for Delphi *FormatDateTime* function.
- **year, month, day, hour, minute, second**: Value interpreted as date, returns the part indicated as a number.

There is an exception, or a modifier that uses a different syntax, and that allows a text to be splitted indicating a specific separator:

- **{xx.N}, {xx_N}, {xx, N}...**: That is, field, separator character, and a number. Returns the token number "N" of the value interpreted as text splitted by the indicated separator. Valid separators are: ':', ',', ';', '=', '_', '-', '/', '\', '|', '!'. If instead of a specific "N" number, letter "L" is written, it returns the last non-empty segment.

17.9 Where can they be used?

Syntax with substitution parameters is accepted in these fields:

- **Templates** of alarm messages.
- **Mapped** values: Destination value.
- **Transformation** with mathematical function $f(x)$.
- Calculated variables of type "**user function**".
- Tags and values in **InfluxDB**.
- External data log **ID**.
- Writing **actions**.

18 Backup copy

Notify can create and restore backups, in automatic or manual mode.

18.1 Backup types

The backup copies made can be of these types:

- **Settings:** Contains only application configuration data (variables, alarms, contacts, general configuration ...)
- **Complete:** In addition to configuration data, it contains historical log data for variables, alarms, and messages sent or received.
- **Incremental:** Contains modified configuration and registry data since the last full copy. To restore an incremental copy, you must have the original full copy in the same folder as the incremental copy.

18.2 Automatic backup

See section 19.10.7-Backup for more information.

18.3 Manual backup

Go to *Settings* → *Global* → *Project* → *Backup* → *Create backup*

First, you will be prompted for the location (folder) where to locate the backup file. If you have set a default folder in the general settings of the application, you can use this, or select another location manually. Then indicate the type of copy to be made (settings, full, or incremental). If you select incremental, note that a previous full copy must exist in the same location. If not, a full copy will be made instead. The process is done in the background and showing a message with the result when the copy is finished.

18.4 Restore a backup

Go to *Settings* → *Global* → *Project* → *Backup* → *Restore backup*

First select the backup file to restore. The home folder is set in the global settings, but you can manually select any alternative location. Next, the list shows the name, date, and size of the files found, as well as the type of backup, sorted by descending date. From this same window you can delete old copies if you think you will not need them. If you delete a full copy, any incremental copies that depend on it will also be deleted. To restore a specific copy, select it from the list with a double click or OK, and then you will be prompted for the desired restore method, which may be:

- **Settings only:** Available for all file types. Only general configuration data, variables, alarms, contacts, etc. are restored, but not the historical data. At the end of the process, Notify verifies the integrity of the historical data, deleting variables and alarms that do not exist in the new restored settings, if applicable.
- **Replace:** Available for full or incremental copies. In this case, an exact copy of the data file is replaced, completely replacing the current one. The result is like go back to the exact same point where the backup was made.

- **Mix:** Available for full or incremental copies. In this case, the copy is restored and the possible historical data (variables, alarms, messages) after the backup date are added. Notify also verifies the integrity of the final data.

The restore process cannot be done while the application is running, so a restart of the application is required, which may be immediate or later if desired. After restart, before loading the new settings, Notify runs the restore process.



It is recommended to enable automatic backups, and use a different drive than where Notify is installed to locate the backup files.

If you have an Internet connection, you can use a cloud storage (Dropbox, OneDrive ...) installing the official client, and then saving the copies of Notify in the synchronized folder.

19 Settings

Although they refer to functions or characteristics seen in previous sections, all the configuration parameters that can be accessed from the main *Settings* window are described here.

Modification of most of these technical parameters is restricted to the administrator user level. May seem there are a lot of configuration parameters, but do not see this as a negative. With the typical default settings, you can start working in a few minutes, and if you need it later, you can only modify or tune the necessary settings.

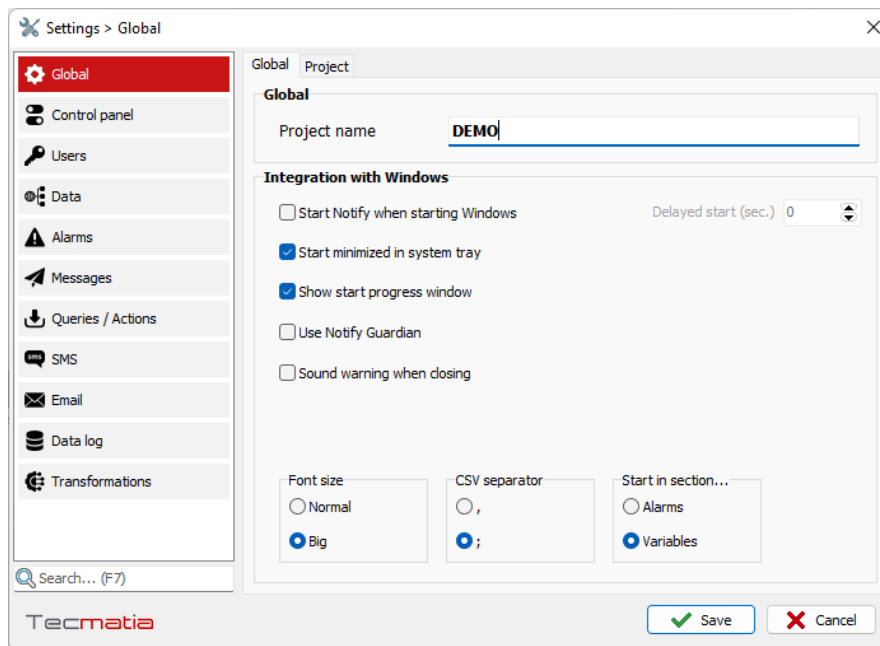


Illustration 33: Main Settings

The main configuration settings are separated into the sections detailed below.

19.1 Global

19.1.1 Global

Global:

- **Project name:** Enter here the text that identifies the project or installation, which will be used in status report messages or alarms.

Integration with Windows:

- **Start Notify when starting Windows:** Specify whether you want the application to start automatically when you start the current Windows session. You can configure a delay time at start-up, to wait for the start and stabilization of other applications that it depends on (OPC server, for example).
- **Start minimized in system tray:** Once you have the project configured and ready, the usual will be to check this box, so that Notify will work in the background, minimized in the Windows system tray, on the right side of the taskbar.
- **Show start progress window:** While starting Notify, a small window in the lower right part of the desktop will be displayed for a few seconds.

- **Use Notify Guardian:** This is an external application capable of monitoring the correct execution of Notify, and ordering its restart in case it *does not respond*.
- **Audible warning when closing:** If the operator closes the application, it is warned by three beeps to emphasize that the closure is not accidental. Enabled by default.
- **Font size:** Affects the main application windows, in variable lists, alarms, etc. Depending on screen resolution, it may be more convenient selecting a large font size.
- **CSV separator:** Will be used when generating CSV files for external editing. Select ',' or ';' as appropriate, depending on the external application that you will use to edit them. For example, ';' works fine with the Spanish version of Excel.
- **Start in section...:** Indicate what section will be maximized when starting Notify, variables or alarms.

19.1.2 Project

Data location:

- **Default:** All settings and internal database is stored in the standard folder assigned by Windows.
- **Folder:** You can select an alternative location for configuration files and database.
- **Open data folder:** Open the selected data path in Windows Explorer.

Backup:

- **Create backup:** Allows you to create a backup of the entire project.
- **Restore backup:** After restoring a previous backup, it will be necessary to restart the application for the changes to take effect.

Export / Import:

- **Export project / Import project:** Option similar to backup, but only for export / import configuration files into a ZIP. Does not include historical data files.
- **Export log files:** Export log files to a ZIP file for technical analysis.

19.2 Control Panel

This panel centralizes the main functions of the application, allowing to quickly enable or disable certain features, or to see at a glance the global status (send alarms, SMS, email, sound, etc.). Please note that these "ON / OFF" switches are a replica of the existing in its corresponding section, so it is irrelevant to use one or the other when enabling or disabling each feature. If you enable an option, but it is not configured correctly or the settings are not complete, you must complete all the necessary parameters in the corresponding section.

19.3 Users

19.3.1 Users

On this tab you can change passwords (administrator and operator) or username (only for operator). Also, you can change the idle time after which the user session is automatically closed. Setting a time of 0 minutes means never log out.

External login (Grafana):

In projects where you use Notify as the data logging system for display in Grafana, it may be useful to share the users already declared there. To do this, activate this option and enter the location of the Grafana database.

External login (OPC):

It is possible to configure Notify to obtain the login information from an external system, typically a SCADA software, if the data is available over OPC-DA.

- **OPC server:** Specify or select the OPC server, using the “...” button.
- **User Item:** Enter or browse to locate the item name that indicates the username (text).
- **Item Level:** Enter or browse to locate the item name that indicates the user level (numeric).
- **Operator level:** Indicate over what numerical value in the "Item level" tag the authenticated user is considered to have operator privileges.
- **Administrator level:** Indicate over what numerical value in the "Item level" tag the authenticated user is considered to have administrator privileges.

Please note that user levels will depend on the design of your SCADA or external application.

It has the button “*Indusoft settings*” that facilitates the configuration for connection to an Indusoft Web Studio SCADA, software we have validated for this functionality. If you want to integrate user identification with other systems, please contact Tecmatia or your distributor for technical support if needed.

19.3.2 Permissions

In this tab you can configure at which user level certain Notify features are allowed, such as closing the application, writing variables, acknowledging or silencing alarms, etc.

19.4 Data

19.4.1 Data source

Data clients:

- **Count "bad quality" as a read error:** If you check this box, when a variable returns a value with status marked as "bad quality", it will be considered and counted as a reading error, for the purposes of displaying the status data, or for activating system alarms. The meaning of a "bad quality" status depends on your server or device, but usually it refers to no connection to a remote device.

OPC UA: Nodetd nomenclature syntax:

It allows selecting the type of nomenclature to use for the item from an OPC-UA server, since there are two different formats commonly used.

19.4.2 OPC-DA server

- **Enable (ON/OFF):** Activate the OPC-DA server. Note that this operation requires administrator privilege in Windows.

Nomenclature of items and folders

- **Path for variables and alarms:** You can use the folder or the name of the data client and reading group.
- **Name of OPC items:** Indicates which text will be used as the name of the OPC variable:
 - *Name:* Default value. The OPC item name is the name of the variable.
 - *Internal ID:* The OPC name will be of the type "vNNN", "aNNN" ,..., formed with a prefix and the internal ID of the variable, alarm, contact, etc. This ensures that the OPC name does not change even if the internal name is changed in notify.
 - *Alias, or name:* In this case, the OPC name will be the *alias* of the variable or alarm in Notify. If it has not been configured, take the name.
 - *Alias, or ID:* Likewise, the OPC name will be the *alias* of the variable or alarm in Notify. If it has not been configured, take the internal ID.
- **Remove accents, Change spaces to '_':** Removes accents and characters that can be problematic in the item names served by OPC, depending on the client.

Permissions

- **External login:** If checked, writing in the username items will be allowed, so that it is possible to login to Notify from an OPC client.
- **Write variables:** Allow or not the writing of variables via OPC, if the source variable also allows it.
- **Enable / disable Notify features:** Allows to activate / deactivate certain functions, such as alarm notification, etc.
- **Modify contacts:** Allow to change contact information, like email or phone number.
- **Execute actions:** By writing a "True" value in the corresponding item.

Other options

- **Save log for writings:** Leave a note in the log files when receiving a new value for a variable.
- **Use source value (do not apply transformation),** for very specific uses. The read/write value from OPC is the original value read from the data connection in Notify without applying transformation rules or possible mathematical operations defined in the variable.

19.4.3 REST API

In this section you can enable and configure the REST API service.

- **Enable (ON / OFF):** Activate the REST API server.
- **Port:** Indicate the listening port for API requests.
- **HTTPS:** Enable security using HTTPS standard.
- **External cert.:** In case of enabling HTTPS it will be necessary to use certificates to guarantee security and encrypt communications. By default, Notify uses internal certificates, generated during the installation process. If you want to use other certificates, check this box, and indicate the path of the files on the *External Certificates* tab.
- **API Keys:** List of valid API keys. They can be read-only or read/write. When you add new keys, they are randomly generated with letters and numbers, but you can edit them to your liking.
- **Save log of writes received:** Records write operations to Notify global log files.

- **Remote connections:** In this tab you can centrally configure the connection settings to remote REST API instances, assigning them a name, which you can then use to make remote writes, transfer local variables to those instances, etc.

19.5 Alarms

19.5.1 Alarms

Global:

- **Send alarms (ON/OFF):** Activate or deactivate whole alarm notification system.
- **Do not generate alarms with "bad quality" values:** It ignores the value of the variable if it is marked with the status "bad quality" for the activation of alarms.
- **Do not notify if there is another active alarm of higher priority ...**
 - ... in the same group or higher
 - ... in a higher group
 - ... in any group

In some cases, when a higher-level alarm occurs, other dependent lower-level alarms may be triggered simultaneously. In this situation, you do not want to receive a batch of messages, but only the global one. To control these situations, configure the alarm priority and places them in group hierarchy, and then check this box to indicate your preference.

Acknowledge:

- **Send confirmation OK response:** Check if you want to receive a reply message with OK when acknowledging an alarm by message.
- **Max time for acknowledgement:** Configure here the default wait times for acknowledgment of alarms based on their priority, before generating a next notification.
- **Acknowledge by message...:** Indicate whether to allow acknowledge the alarms by sending a message, and if so, from which contacts it should come. If not allowed, pending alarms can only be acknowledged in the application UI.

19.5.2 Notification

Retries (on error):

- **Maximum tries** Indicate the number of times to try to process an outgoing message on a delivery error. It affects all types of messages. It counts a delivery error if the message system is working (it connects to the email server, modem is registered in the GSM network, etc.) but the sending of the specific message fails.
- **Abort message if not sent for N hours:** If the messaging service is not working for a long time, the messages will be queued in the outbox pending delivery. With this parameter you can specify a time after which the message is aborted without sending.

Desktop popup with active alarms:

- **Show:** Check if you want a desktop pop-up message to be displayed when an alarm occurs. It is possible to limit for a minimum alarm priority.

- **Show system alarms:** Checking this box will display the desktop pop-up message only for enabled system alarms, that is when there is any problem in the operation of Notify (message delivery failure, Internet unavailable, etc.).

Sound:

- **Play sound when an alarm is activated:** Allows play a sound when a new alarm occurs. You can limit to only the first active alarm, or from a minimum priority. The sound that is played and the volume of this will depend on your operating system settings.

19.5.3 Message templates

Although it is possible, it is not necessary to define a custom message for each alarm. Instead, the usual way will be to configure a generic template that, using substitution parameters, can be used for all alarms. This configuration is separated into different templates, one for each message type and alarm event (activation, deactivation or acknowledge).

To edit a template, select the desired message type and event. It displays an edit window like this. Enter the desired text and use the "Insert variable..." option to insert a substitution parameter at the cursor position.

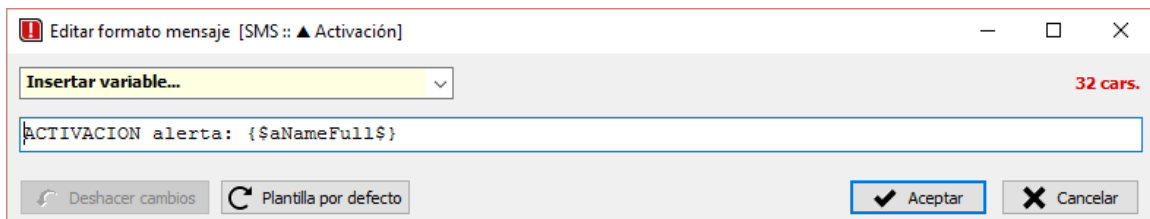


Illustration 34: Editing a message template

When editing messages where length is important (SMS for example), a counter of characters used is displayed on the screen. Note that the actual length of the message will be the result of substituting the parameters for the actual values for each case, so use this data only as indicative.

You can return to the default standard template, using the available button. You can also revert to the last saved version, using the *Undo changes* button.

For emails, in addition to the customizable template in plain text mode, there is a new pre-established **HTML format** that includes all the useful information and is more attractive and easier to interpret, also on mobile devices. We recommend using this new format for emails.

Alarma OK	
Nombre	Error de registro en BD externa
Carpeta	[Sistema]
Prioridad	Normal
Activación	03/01/2020 9:46:39
Desactivación	03/01/2020 9:47:37
Tiempo activa	58s

Illustration 35: Example of HTML email template

19.5.4 Suffixes when linking alarm name

In the same *Message Templates* tab, you can configure the texts that will be used to get the alarm name when it is linked to the variable with the option *Hi*, *Lo*, *Out* or *Fail*. These shortcuts are intended to facilitate the creation of “High value”, “Low value”, “Out of range”, or “Fail” alarms, respectively. Note that texts configured here are added as a suffix to the name of the variable.

19.6 Messages

19.6.1 Status report

General:

- **ON / OFF:** Enables the sending of scheduled periodic status messages. It does not disable status messages on request (sending a message to Notify).
- **Enable:** Check the messages types you want to enable.
- **Include list of active alarms:** Check this option if you want to add information about the active alarms to the status message. For short messages (SMS) only the number of active alarms is added; for long messages (email) a detailed list of alarms is included. In both cases, the result is restricted to alarms with a level greater than or equal to that indicated.
- **Used disk space:** Check this option to add this information to the message. If the space used is greater than the configured alarm value, the data will be highlighted in red.

Timetable:

- Schedule when the status report will be sent.
- **For contacts with active hours, it is sent when they start working:** Checking this box, contacts with a configured schedule other than 24-7 will receive a status message at the beginning of their working time, independent of the global time scheduled.
- **Assigned contacts:** The contacts to which the status report will be sent. The assignment of these contacts can be done either from the main *Contacts* section, or from this shortcut.
- **Preview:** Used to show how the status message will look, depending on the type of message available.
- **Send now...:** Use this button to send a status message now to a contact in the phonebook. It can be useful to test the format or the messaging system.

19.6.2 Messages from external sources

Send messages generated in an external application is supported, linking with an SQL database.

- **ON / OFF:** Enables the external messages service. When enabled, a new icon will be displayed in the upper status bar, to check if there is any warning in the service. It also has a specific section in log files.
- **Database:** Configure the connection to the external database. Select the correct driver first, as the rest of the parameters depend on it.
 - *Ora*: Oracle Database.
 - *MSSQL*: Microsoft SQL Server.
 - *MySQL*: MySQL, MariaDB.

- *PG*: PostgreSQL.
 - *FB*: Firebird, Interbase.
 - *SQLite*
 - *ODBC*: Other databases that have ODBC access.
- **Check every...:** Enter the interval to check for new messages, as a time literal.
 - **Allow sending only to contacts who are in the phonebook.**

Query of messages to send:

- **Query:** Add here the SQL query or queries (SELECT) that will return the new messages to send. It is the only required SQL. You can use the following dynamic parameters:
 - *:dateTime*, *:date*, *:time* → Current date time (full, only day, or only time, respectively)
 - *:lastMsgID* → Numeric identifier of the last message processed
 - *:lastMsgTime* → Date of the last message processed.

The query should return these fields:

- *type* → Type of message ('SMS', 'Email')
- *id* → Numeric identifier of the message.
- *msg_time* → Message creation date. By default, now.
- *schedule_time* → Shipping date. By default, now.
- *limit_minutes* → Deadline for delivery, in minutes.
- *limit_time* → Deadline for delivery.
- *priority* → Message priority. Text or number, 0 = 'Lowest', 1 = 'Low', 2 = 'Normal', 3 = 'High', 4 = 'Highest'. By default, 'Normal'.
- *param1*, *param2* ... → Other data to bind in other SQL if necessary.

For SMS messages:

- **text*, *subject* → Text.
- *sender* → Sender.
- **phone*, *recipient* → Addressee. Field 'recipient' accepts phone number or name of a contact, while 'phone' needs a phone number.

For email:

- **text*, *subject* → Email subject.
- *body* → Email body.
- **email*, *recipient* → Addressee. Field 'recipient' accepts email address or name of a contact, while 'email' expects only an email address.
- *cc* → Recipients in copy.
- *bcc* → Hidden copy recipients.
- *from_email*, *sender* → Sender (email).
- *from_name* → Sender (name).
- *content_type* → Content type ('text/plain', 'text/html'). By default, 'text/plain'.

Fields marked with * are required.

- **Before:** SQL statements to execute before executing the main query.
- **After:** SQL statements to execute after executing the main query.

Execute after sending message

In addition to the dynamic parameters valid for any SQL, you can use these that refer to the sent message:

- *:dispatchTime* → Delivery date.
- *:dispatchResult* → Result of the process, in text.
- *:dispatchOk* → Result of sending correct or not (Boolean: True = Ok, False = Error).
- *:dispatchOk01* → Result of sending correct or not (Numeric: 1 = Ok, 0 = Error).
- *:msgID* → Numeric identifier of the message, if available.
- *:msgTime* → Date of the message.
- *:msgText* → Message text.
- *:msgSender* → Sender, if available.
- *:msgRecipient* → Addressee.
- **If Ok:** SQL statements to execute if the message has been sent successfully.
- **If error:** SQL statements to execute if the message could not be sent.
- **Ever:** SQL statements to execute in any case. Executed after the previous ones.

Periodically

Optional, allows you to execute one or more SQL statements at regular time intervals. Enter the interval as a time literal and use the available dynamic parameters if needed.



Notify allows to act as a gateway to send messages of external origin. This is interesting for centralizing the sending of messages generated by different applications.

19.6.3 Translation

Although the Notify interface is only available in Spanish and English, this section allows you to translate the texts that can appear in the standard notification templates. In this way, the end user will receive the messages in their language, independent of the language of the application. It has preconfigured translations for some languages, but you can create your own translation.

19.7 Queries / Actions

- **ON / OFF:** Globally enables the queries requested by incoming messages.
- **Enable:** Check the message types for which you want to enable for queries / actions.
- **Allow from contacts that are not in the phonebook:** By default, if you do not check this box, queries or actions will only be enabled for senders (phone, email) who are in the Notify contact list.

- **Ignore messages older than N minutes:** Allows you to specify a time from which the incoming message is ignored as it is considered to have expired. The date taken is that of the creation of the original message.
- **Include query command in reply SMS:** In queries, for long messages (email) the read command used is always included in the response. For short messages (SMS), where the length is relevant, it can configure whether to include it or not (*Never, Only if it fits, Always*)
- **Send confirmation OK answer:** Enable if you want to receive a confirmation message on successful completion of an action. You may be interested in disabling it if the messages have a cost.
- **System queries / actions:** Opens the edit window for *Predefined system queries or actions*.

19.8 SMS

19.8.1 SMS

Format:

- **Header:** You can enter a text here that will always be added to the beginning of all outgoing SMS.
- **Footer:** You can enter a text here that will always be added to the end of all outgoing SMS.

Send:

- **Primary system:** It is possible to simultaneously enable the GSM modem and an online SMS gateway. In this case, select here which will be the main delivery system. Notify will always use this channel as a priority to send SMS and will only switch to the secondary channel when the main one fails.
- **Default international prefix:** When using a GSM modem, the encoding format of the telephone number when sending SMS in some cases depends on the network operator. Some require the international prefix in "00xx" format, others only in "+xx" format, while others allow both. Select in this field the international prefix that will be used by default for contacts that do not have one defined in the phone number, in the format required by your operator. If you experience problems with sending (in Notify it appears as sent OK, but you do not receive SMS), try changing the prefix format. In the case of SMS online gateways, this prefix is used to complete the telephone numbers of the contacts that do not include it, but the format of "00xx" or "+xx" is not relevant, since Notify will use the one that each supplier requires.
- **Ignore limit if alarm priority > =...:** If you set a daily or monthly message limit, when the limit is reached, Notify stops sending any message, except if you use this option to ignore this restriction on high priority alarms.

Ignore incoming messages:

- **If not a contact from the phonebook.**
- **If the text matches a pattern:** Check this box if you want to specify a text, an MS-DOS pattern, or a regular expression with which to compare the content of the incoming message.

19.8.2 GSM modem

- **Send SMS (ON / OFF):** Enables the sending of short SMS messages through the GSM modem.

Serial port:

- **COM:** Select the serial port to which the GSM modem is connected, either through the COM port number or the selection combo that recognizes the available ports. Use the *Refresh* button if you do not find a installed port.
- (...): This button accesses the serial port properties configuration, specific to the operating system, where you will find these parameters:
 - *Bits per second (bps, baudrate):* Communication speed. Set a value supported by your device. Typically, modern modems automatically adapt to the selected speed without the need for external configuration and allow high baud rates or bps to be used.
 - *Data bits, Parity, Stop bits, Flow control:* Consult the technical documentation of the GSM device for the correct configuration of these parameters, or keep the default values in case of doubt.
- **Reconnect by name:** Usually the connection will be made directly to the COM port with the indicated number. By checking this box, if there is no connection to the COM port number, Notify will try to search the list of ports for another one with the same name, and if found, it will connect to that COM port number. This can be useful for USB devices that sometimes when physically reconnecting the cable assigns a different port number.
- **AT commands driver:** Select the driver that best suits the manufacturer and model of your device. When in doubt, use the default *Generic GSM Modem*.
- (...): This button shows the list of AT commands for the selected driver for technical detail.
- **Timeout - R/W (ms):** Maximum time for data reception (R) and transmission (W). In some modems configured with a low baudrate, it has been found that it is necessary to use high values of timeout (2000 ms) for the correct sending of a long SMS. At high baud rates, times may be shorter. If you experience problems with sending some SMS, try changing this parameter.
- **Timeout - Send SMS (sec.):** Maximum time for the SMS sending operation. A typical value for the time to send an SMS is on the order of 2 to 4 seconds. However, this time varies depending on the coverage and availability of the GSM network, so it is advisable to set a higher value, around 10 seconds.
- **Enable debug log (AT commands):** Activates the log of AT commands sent and received. Useful to analyse possible communication problems, or the correct response to the AT command.

SMS:

- **Encoding:** Select the suitable encoding format according to the GSM 03.38 standard:
 - *7 bits (GSM):* The maximum length of the SMS is 160 characters. Although Notify tries to adapt the content if necessary, it can give problems with accents or some special characters. It supports using an extended character set.
 - *8 bit (ANSI):* Each character is encoded with 8 bits, so the set of possibilities is greater, but the maximum length is reduced to 140 characters.
 - *16 bits (Unicode):* Allows you to encode virtually any character in any language. The maximum length is 70 characters.
 - *Auto:* If you select this option, Notify will select the suitable encoding system for each SMS sent, depending on its content (if it includes Unicode characters, or accents...). So, the maximum length will be variable, from 70 to 160.
- **Extended character code:** Option available for 7-bit encoding system. The GSM standard defines several character sets greater than the basic one for some languages where the basic 7-bit set is not sufficient. Notify implements those corresponding to *Spanish* and *Portuguese*. When using

one of them, the maximum length of the SMS is 157 characters, but it allows the use of accents, and other characters such as ñ, ç, etc. If you select the *Auto* option, Notify will use one or the other depending on the characters found in the SMS to be sent.

- **If the text exceeds one SMS:** Select what action to take when the length of the SMS to be sent is greater than the maximum allowed, according to the encoding format seen in previous points.
 - *Keep:* The SMS is sent as is to the GSM network. The result and the pricing applied will depend on your GSM operator.
 - *Truncate:* The SMS is always cut to the maximum length. It guarantees that a single SMS is always sent and charged.
 - *Split:* In this case, it is Notify that cuts the long message into several SMS of the maximum length and sends them all in order, as if they were individual SMS.

SIM:

- **PIN:** Although it is advisable to use an unlocked SIM, it is not required. Enter your card PIN here if you prefer to keep it blocked. When initializing the modem, Notify checks the need for a PIN and transmits it to you if necessary. If it fails, no retries are performed, to avoid a possible permanent blocking of the SIM card.
- **Phone number (optional):** You can enter the local phone number of the SIM card here. It is not necessary for sending and receiving SMS, it is only informative.

Message limit:

- **Max. day:** Check the box and enter a value if you want to limit the number of outgoing SMS per day.
- **Max. month:** Check the box and enter a value if you want to limit the number of outgoing SMS per month or billing cycle. You can indicate which day of the month to reset the counter.



Notify allows you to set maximum limits on sent messages, in order to control a possible high value in unstable installations.

Receive:

- **Receive (ON / OFF):** It enables the reception of SMS messages and calls through the GSM modem.
- **Automatic (recommended):** In this mode, when the modem detects a new incoming SMS, it notifies the application to be processed. It is the recommended mode if your GSM device supports it.
- **Check every N seconds:** In this mode, an active check is made periodically by consulting the modem if any SMS has been received. Use the method that works best for your GSM network and modem.

19.8.3 SMS gateway (online services)

Allows sending SMS through online services (requires Internet connection).

SMS gateway:

- **Provider:** Select here the SMS online service provider, from the list of supported by Notify. Note that you must contract an account directly with the provider, which will provide you with the necessary configuration data to enable delivery from Notify.
- **Encoding:** Select the encoding format for the SMS. Not all services have support for all formats but Notify will prepare the text of the SMS before sending if necessary.
 - *7 bits (GSM):* The maximum length of the SMS is 160 characters. Notify tries to adapt the content if necessary, but in some cases, it could give problems with accents or some special characters.
 - *16 bits (Unicode):* Allows you to encode virtually any character in any language. In contrast, the maximum length of each SMS is 70 characters.
- **If the text exceeds one SMS:** Select what action to take when the length of the SMS to be sent is greater than the maximum allowed:
 - *Keep:* The SMS is sent as is to the provider. The result will depend on the provider rules.
 - *Truncate:* The SMS is always cut to the maximum length. It guarantees that a single SMS is always sent and charged.
- **Parameters:** When selecting a provider, you will see in the parameters panel the values that you will need to configure for that specific provider. If a password or secret token is needed, it appears in a separate and hidden content field to ensure privacy. This data is stored encrypted in the configuration files.

Message limit:

In the same way as for the GSM modem, it is possible to configure a limit of messages sent per day or month.

19.9 Email

You will need an Internet connection to be able to use the email sending and receiving service. If you do not have your own email account or server, you can create a free account with several common providers (Gmail, Yahoo, Outlook.com, etc.). Some of them require a previous configuration of the account to be able to be used by third-party applications. This is the case with Gmail, for example. Check your provider's documentation to see if you need to enable any special permission to be used with Notify.

19.9.1 Output (SMTP)

Server:

- **Send emails (ON / OFF):** Globally activate or deactivate the sending of emails.
- **Server / Port:** Host name and port where the SMTP server resides. You can use the "..." button to select the quick configuration of the main free email providers (Gmail, Yahoo, ...)
- **Security method:** Select the level of security required by your email provider.
- **Keep connected:** By default, Notify performs a connection and disconnection process to the outgoing server for the sending of each batch of emails. This process requires a little time, depending on the server and the Internet connection. Usually this time will not be important, but if your system generates a high number of outgoing emails, you may want to keep the connection always open and speed up delivery.

Sender:

- **Name:** Indicate the descriptive name that will appear as the sender of the emails sent in the destination mail client.

Authentication:

- **Does not require authentication**
- **Username / Password** If authentication is required.

Hidden copy:

- **BCC:** Allows you to send a copy of all emails sent to one or more email addresses. Use a comma (',') to separate more than one.

Message limit:

In the same way as for the SMS, it is possible to configure a limit of messages sent per day or month.

19.9.2 Input (POP3 / IMAP)

Server:

- **Receive emails (ON / OFF):** Globally enables or disables the reception of incoming emails.
- **Server / Port:** Host name and port where the server resides. Use the "... " button to select the quick configuration of the main free email providers (Gmail, Yahoo, ...)
- **Protocol:** Select the protocol for receiving emails (POP3 or IMAP) as available from your provider and your preference. IMAP is a more modern protocol than POP3. However, for use with Notify you will not notice big differences between one or the other.
- **Security method:** Select the level of security required by your email provider.
- **Check every ...:** Emails are acquired by active consultation, that is, Notify connects periodically and checks for new incoming emails. Set a reasonable email acquisition time, based on acceptable response time, to avoid excessive strain on the network or your email provider. Please note that some free email providers may block access if they understand that the service is being misused, since the email system has not been designed for immediate responses. The concept of abusive use is relative, but it will be defined by your provider. We do not recommend times shorter than 30 seconds, except for testing or validation.

Emails to download:

In your email account you may receive SPAM emails, or with large attachments ... However, the emails relevant to Notify (queries / writes / actions...) will usually be short messages. To avoid possible blockages due to excessive download time of irrelevant emails, especially in slow Internet connections, it is possible to configure some filters:

- **Only plain text emails:** If you check this box, simple emails will be downloaded in text mode. Not those formatted in HTML or rich text. Note that if you check this option, you must use plain text in the emails you send to Notify for queries or actions, correctly configuring the email client you use.
- **Limit size:** Avoid downloading emails larger than the indicated size. In practice, avoid emails with attachments.

- **Delete messages from server after downloading:** Check this option if you do not want to keep a copy of the emails received on the server. It may be convenient if you have a limited mailbox, but it does take a bit more time for the receiving process.

Authentication:

- **Use SMTP credentials:** Usually the user identification to download and send emails is the same.
- **Use Username/Password:** If the authentication is different than for SMTP.

Ignore incoming messages:

- **If not a contact from the contact list.**
- **If the text matches a pattern:** Check this box if you want to specify a text, an MS-DOS pattern, or a regular expression with which to compare the content of the message to ignore it.

19.10 Data log

19.10.1 Internal

- **Variables (ON / OFF):** Globally enable the internal historian of variables (the alarm historian is always active). By default, it is enabled.
- **Limit time (days):** Typically, your license will limit the maximum days of data available. However, you can limit to a lower value for optimizing disk space or other reasons.
- **Limit space (MB):** It does a similar function to the previous point but limits it to a specific disk space. When overcoming it, the oldest data is eliminated.
- **Log files - Limit time (days):** Allows you to limit the maximum number of days that log files are kept. By default, 60 days.

19.10.2 SQL: Variables

If your license enables this feature, Notify allows the insertion of data in an external database of type RDBMS (SQL). This section requires a minimum knowledge of databases and SQL for the correct configuration of the connection to the database and the tables where you want to save the information.

- **ON / OFF:** Globally enable the historian of variables in the external database.
- **Database:** Configuration of the connection to your external database. Select the correct driver first, as the rest of the parameters depend on it.
 - *FB:* Firebird, Interbase.
 - *MSSQL:* Microsoft SQL Server.
 - *MySQL:* MySQL, MariaDB.
 - *Ora:* Oracle Database.
 - *PG:* PostgreSQL.

- **OBDC:** Other databases that have ODBC access.

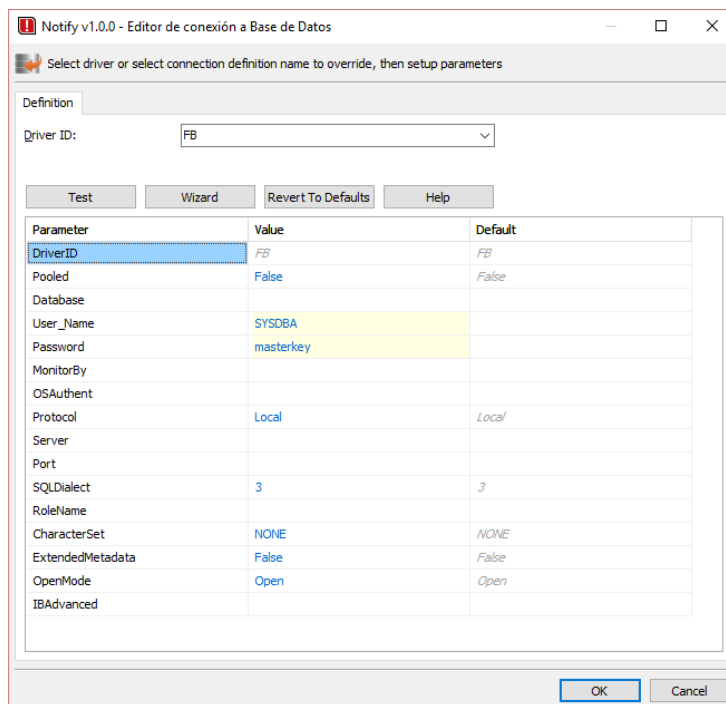


Illustration 36: External database connection configuration

Fill in the data required by the selected driver (server or host, port, database, username, password ...). Use the "Help" button or your database server documentation for help with the necessary settings. You can request technical assistance from Tecmatia if this information is not enough.

If necessary, use the "Revert to Defaults" button to return to the default configuration parameters.

When the configuration is finished, use the "Test" button to verify that Notify can make the connection correctly. If you are done, press OK to save the current settings.

- **SQL (Value):** SQL statement that will be executed to insert a new value in the database.
 - **SQL:** Enter in this tab the text of the SQL to be executed, using the necessary substitution parameters and macros if needed from those available.

Parameters: Substitution parameters are used in the values section of the SQL and will be replaced by a specific value at the time of processing the SQL. Available:

- **:id** → Internal ID of the variable (Notify).
- **:idExt** → External ID configured in the variable.
- **:name** → Name of the variable.
- **:alias** → Alias or short name.
- **:serverName** → Name of the data server from which the variable is obtained.
- **:serverGroup** → Group name within the data server.
- **:serverPath** → Full path of the data source (server + group)
- **:folderName** → Name of the final folder in the variable tree.
- **:folderRoot** → Name of the root folder in the variable tree.

- *:folderPath* → Full path in the variable tree.
- *:dateTime* → Current date and time.
- *:date* → Current date.
- *:time* → Current time.
- *:value* → Current value of the variable.
- *:valueRound* → Value rounded to the nearest integer.
- *:valueTrunc* → Value truncated to the nearest lowest integer.
- *:valueRoundDec* → Value multiplied by the number of decimal places configured for the variable and rounded to integer.
- *:valueText* → Current value in text.
- *:valueUnits* → Units of the variable, in text.
- *:customField1..3* → User custom field.

Example: If the current value of the variable is "12.78 m3/h", and it is configured with two decimal places for display, these would be the values that would be used in the SQL:

- *:valueRound* → 13
- *:valueTrunc* → 12
- *:valueRoundDec* → 1278 (= 12.78 * 10²)
- *:valueText* → "12.78"
- *:valueUnits* → "m3/h"

Macros: Macros are also texts to replace when processing SQL, but instead of values to insert they are typically used for table or field names. Available:

- *&id* > Internal ID of the variable (Notify).
- *&idExt* → External ID configured in the variable.
- *&name* → Name of the variable.
- *&alias* → Alias or short name.
- *&serverName* → Name of the data server from which the variable is obtained.
- *&serverGroup* → Group name within the data server.
- *&serverPath* → Full path of the data source (server + group)
- *&folderName* → Name of the final folder in the variable tree.
- *&folderRoot* → Name of the root folder in the variable tree.
- *&folderPath* → Full path in the variable tree.
- *&customField1..3* → User custom field.

Example: Let's suppose a variable with alias "MY_VARIABLE", in the folder "FOLDER_1", with external ID "115", and current value is "3.14". The following SQL statement:

```
UPDATE TABLE_DATOS_&folderName SET &alias=:value, DATE=:date WHERE ID=:idExt
```

It will be executed as:

```
UPDATE TABLE_DATOS_FOLDER_1 SET MY_VARIABLE=3.14, DATE='04/01/2018' WHERE ID=115
```

- *Parameters / Macros*: In these tabs you will see the substitution parameters and macros, respectively, that the SQL contains. It allows you to fine-tune the type of data or size of the associated field in the database. This information is covered automatically for the parameters described above, so on rare occasions it will be necessary to modify it.
- **SQL (Periodic)**: Optionally, it allows an SQL statement that will be executed every time interval. The configuration is like the one described in the previous point, but you will not have data for a specific variable, since in this case there is no reference to any. Also enter the interval the SQL should be executed, in seconds, minutes, hours or days. One possible utility is to update a date, counter, or status table.
- **Insert NULL before start**: Before first record, Notify will insert a *Null* value for each variable as a “start mark”. This value is used in some third-party systems to mark in which time frames there is no data available.
- **Insert NULL after finish**: Similar to the previous, but inserting a *Null* for each variable after the last record, either because it is disabled or because Notify is closed.
- **Save current value at 00:00**: Forces a data record at each day change to ensure there be at least one value per day.
- **Only if there is no value in the day**: In conjunction with the previous parameter, save the value at 00:00 only if a data does not already exist for the entire day that ends.

19.10.3 SQL: Alarms

In addition to the variable data historian, and independent of it, Notify allows data relating to the activation or deactivation of alarms to be recorded in an external database.

- **ON / OFF**: Globally enable the alarm status change historian in an external database.
- **Register in a different DB**: Check this box if the database where the alarm events are to be recorded is other than that of variables, or if you have not enabled this.
- **Database**: If it is a different connection, configure here the parameters for accessing the alarm database.
- **SQL (Activation) / SQL (Deactivation) / SQL (Acknowledge)**: SQL statements that will be executed for each alarm when it is activated or deactivated. The configuration is the same as the one described above for Variable SQL, except that the available parameters and macros vary slightly. Different from SQL variables are:
 - *:dateTimeAct* → Date and time of alarm activation.
 - *:dateAct* → Date of alarm activation.
 - *:timeAct* → Time of alarm activation.
 - *:dateTimeDes* → Date and time when the alarm was deactivated.
 - *:dateDes* → Date of deactivation of the alarm.
 - *:timeDes* → Alarm deactivation time.
 - *:act01int* → Active alarm value in format 0 or 1 as number.
 - *:act01str* → Active alarm value in “0” or “1” format as text.
 - *:act01bool* → Active alarm value as Boolean data type (True / False).

- **SQL (Periodic):** Optionally, as for the variables database, the configuration of an SQL statement that will be executed periodically.

Note that you can configure external historian independently between variables or alarms or use only the SQL you need. If you do not require any SQL to run, just leave the text blank.

19.10.4 SQL: Custom fields

Allows you to configure the name of up to 3 custom fields, which then appear in the individual configuration of variables or alarms to assign values to them. these fields can be used freely in the SQL described above.

19.10.5 InfluxDB

If your license enables this feature, Notify allows data ingestion into an external InfluxDB database, of type TSDB (*Time Series Database*). This section requires a minimum knowledge of InfluxDB for the correct configuration of the connection to the database and the data structure to save the information. Being an open source project, it has very complete documentation on the official InfluxData website. We will cover here only a list of the necessary configuration data, without going into depth in its description or explaining the operation or syntax.

General:

- **(ON / OFF):** Globally enable external historian in InfluxDB.
- **Database:** Required. Connection data to the database:
 - *Server:* Name of the server or IP where the InfluxDB engine resides.
 - *Port:* TCP port number within the server.
 - *HTTPS:* Enable HTTP security if needed and enabled on server.
 - *GZIP:* Compress the data sent to InfluxDB in *GZip* format. Useful for high data rates over the Internet or remote connections. Not recommended for a local database.
 - *Database:* Database name.
 - *User / Password:* With sufficient permissions in the database.
 - *Test:* Use this button to test the configuration and check if database is accessible.
- **Global tags:** Optional. Enter here, if necessary, the tag keys and values that will be applied to all the data recorded from this instance of Notify, that is, all variables or alarms with recording enabled. You can enter up to 2 global tags with their name and value. The main utility is to identify the source instance, in case of sharing the same database among several.
- **Fixed tags:** Optional. It allows configuring up to 6 tag keys, whose values will be specified in each variable or alarm. You can indicate whether they are *required* or not (*optional*). You can also select whether the labels will be included only in the standard Notify record by checking the *Internal* box.

Record types:

It is necessary to declare in this table the types of records that will be applied to each variable or alarm. Basically, it defines the data type and location in the InfluxDB metrics.

- **Name:** Required. Name of the type of record, which will be linked in the variables or alarms.

- **Measurement:** Required. Name of the metric in InfluxDB. You can specify more than one measurement for the same type of record, separating them with commas, in this way the same data will be saved in several different measurements.
- **Precision:** Time precision to apply to the data. Default value is 1 second. Use a format like:
 - "m", "h", "d" for minute, hour, or day.
 - "3m", "2h" ... to round the date to the lower multiple.
 - "10m+", "h+" ... to round the date up to the higher multiple.
- **Min. time:** Minimum time between consecutive samples. At least this time is expected before inserting a new sample from a particular series.
- **Max. time:** Maximum time between consecutive samples. If there are no new data to record, because there is no variation in value, a sample is inserted every this time to ensure that there is at least some data periodically..
- **Value field:** Optional. Field name to store the value in the measurement. By default, "value".
- **Ret. Policy:** Optional. Name of the "*retention policy*" in which the data will be located. See InfluxDB documentation for more information. The same type of record can save data in more than one retention policy, indicating them in this field separated by commas.

To edit the table you have the options of adding / deleting / duplicating a record manually, and copying / pasting locally or from Excel.

Standard data log:

Allows you to activate and configure a record system in InfluxDB with a preconfigured internal architecture with all the available data of the variables and alarms that have activated the external registration in InfluxDB.

- **Instance:** The recorded data have a special label that identifies the Notify instance, useful in the case of centralizing data from multiple instances in a single database. You can check this box if the InfluxDB database is unique or shared, to speed up the data recording and synchronization process.
- **Retention policy:** Allows you to assign a specific retention policy for the metrics in this standard record. Note that the data is saved in InfluxDB with internal timestamp to zero, so the retention policy cannot have a maximum duration configured.
- **Notify standard data log:** Enable here the options that interest.
 - **Alarms: Current state:** It maintains in the *_Notify_AlarmSummary* measurement the current status of the alarms with the information available for each one.
 - **Alarms: Event history:** It records in the *_Notify_AlarmHistorian* measurement the activations, deactivations and acknowledgments of the alarms.
 - **Alarms: Graphic history:** It records the active or inactive status of the alarms in the *_Notify_AlarmGraph* measurement to facilitate a graphical representation.
 - **Variables: Current value:** It keeps in the *_Notify_VarSummary* measurement the current value of all the variables with the information available for each one.
 - **Variables: Connections state:** Maintains the current state of all configured data connections in the *_Notify_DataClients* measurement.

- **Variables: Samples/day:** Records in the measurement `_Notify_VarDaySamples` the number of samples recorded for each variable and day, for the purpose of statistics and control.

19.10.6 FlyThings

Notify can record data on the FlyThings IoT platform (<https://www.flythings.io/>)

In this tab the parameters of access to that platform are configured.

19.10.7 Backup

- **Destination folder:** Indicate the path where Notify will save the backups made in automatic mode.

Automatic:

- **ON / OFF:** Enable or disable automatic backups.
- **Full backup:** Indicate the day of the week and time to perform a full backup (contains all settings and historian data and does not depend on previous copies).
- **Incremental backup:** Indicate the period in hours to perform an incremental copy (contains only the data modified since the last full copy). Minimum 2 hours, maximum 24 hours.
- **Max full backups:** When you make a new full copy, old copies are automatically deleted, up to the maximum number of copies indicated. This setting does not affect creating a manual copy.

19.10.8 Disk space used

This panel displays information about the disk space currently occupied by the different Notify components, as well as the global space occupied on the drive where the application is installed.

If you are experiencing low free space problems, you can modify the settings for each section to limit the space occupied. For example, limiting the internal historian, log days, number of backup copies, etc.

19.11 Transformations

In this section you will find the centralized configuration of reusable transformations for variables.

19.11.1 Point curves

A point curve defines an X-Y transformation function from a series of known points, performing a linear interpolation for intermediate points. It can be used by assigning it as a transformation in the acquisition of variables, so that "x" represents the original value read, and "y" the result.

- **Curves:** A list of curves defined with a name. You can select an existing one, add a new one, or delete. Each curve, in addition to the name, allows you to configure the number of decimal places to display the values, the measure units, or what to do when the input "x" value falls outside the range configured in the graph.
 - **Extrapolate:** Extrapolate the value of "y" as a linear continuation of the end of the graph.
 - **End value:** Returns for "y" the last value set at the end of the graph.
 - **Null:** Returns "null" or error in reading.
 - **=X:** Returns the same input value.

- **=0, =-1:** Returns a particular value.
- **Points:** Enter here the list of X-Y points that determine the graph to apply. Adding new points prompts you for the value of X and Y. You can edit the Y value of existing points directly in the list, but not the value of X. As you modify the points you will see the resulting curve reflected in the graph.
- **Test:** It allows to simulate what the result would be for a specific value of X. The exact X-Y test point is also reflected in the graph.

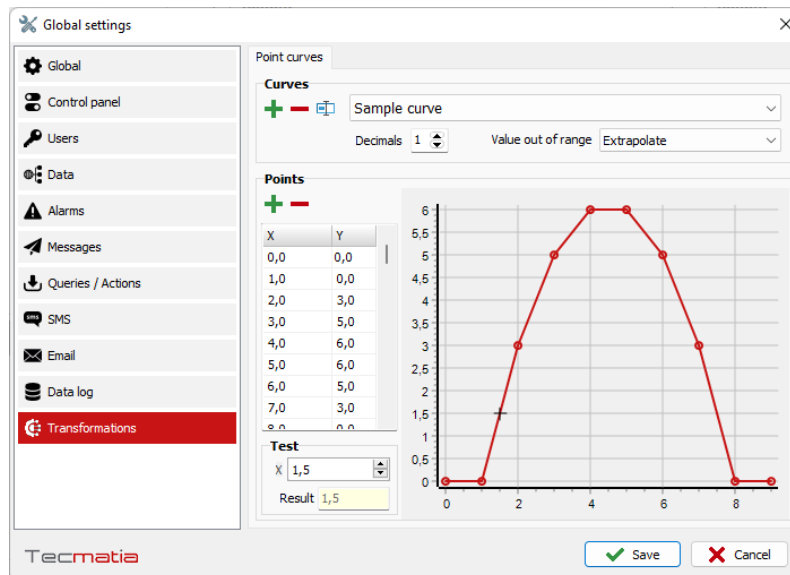


Illustration 37: Point curves.

19.11.2 Timetables

It allows the configuration of two types of calendars or time tables:

- **Periods:** In this type of calendar, a number of periods (from 1 to N) is defined, to which a color can be assigned to facilitate visual configuration. Then the timetables are configured indicating the date of application and the assignment of a period number to each of the 24 hours of the day.

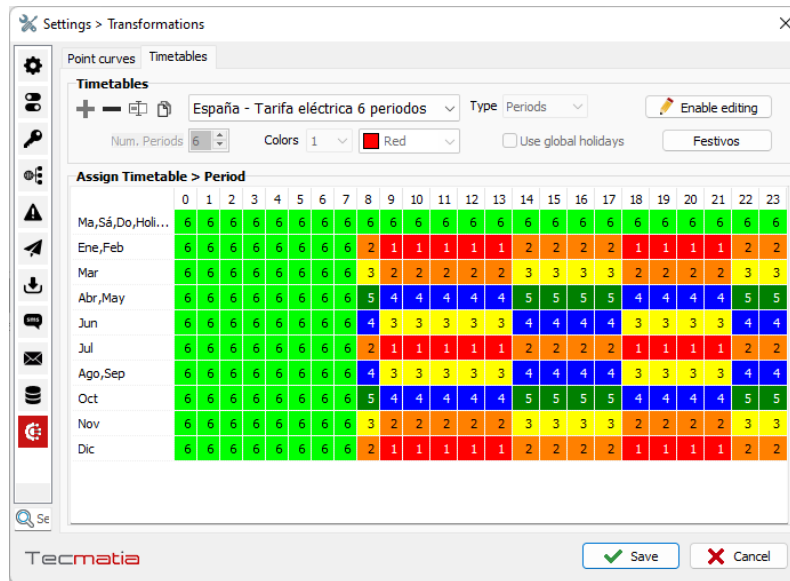


Illustration 38: Periods timetable.

- **On/Off:** In this case, only the list of active intervals is defined, being inactive at any other time that does not match any defined interval.

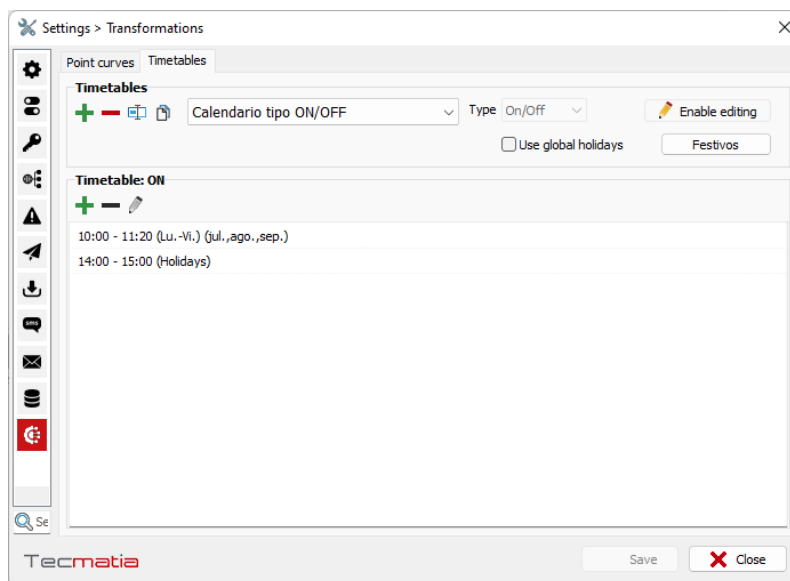


Illustration 39: ON/OFF timetable.

Both types of timetables have in common that they must be saved with a unique name, and allow the individual configuration of holidays or use the global list. In this case, the same holidays configured in the *Contacts* section are used.

20 Log files

Since Notify is conceived for the supervision of alarms in process that could be critical, the application has a detailed log system, in which the main events and incidents are recorded, as well as the status of all the software modules.

Any authenticated user (operator level) can view the log files. Select the module and date from which to obtain information at the top. Use the *Find* or *Locate* functions of the context menu (right mouse button) to speed up the monitoring of files.

It is possible exporting all the log files to a ZIP file, useful to be sent to a technician to investigate any possible incident. You will find this option in *Settings* → *Global* → *Project* → *Export log files*.

As it is a critical application and operates in the background, usually unattended, any internal error of the application or unexpected external errors will not be shown by a message on the screen, which could lead to blocking situations while waiting for user interaction, but it is captured internally and written into the log files. Instead of the message on the screen, an activation of the system alarm "*Notify - Internal error, see log files*" is generated. This alarm can be notified to any contact in the same way as any other alarm. In the email message, a file with relevant data of the incident is attached, which can be sent to the technical service if necessary.

21 Export / Import settings

Notify allows the configuration of a project on a PC other than the final installation, even without having a license (DEMO mode). To facilitate the transfer of the settings to the final computer, you have options to *Export project* and *Import project* in the *Global > Project* tab of the main *Settings*. This tool encapsulates all the needed data in a ZIP file, easy to move.

Note that on the final computer where you are going to install the project, you must have a valid license or enable *Free* edition to be able to run it.

You can use this same export / import tool to make backup copies of the current settings.



Use the Export / Import Project tool to transfer a configuration from a PC other than the final installation, or to make backup copies of the current settings.

22 Keyboard shortcuts

In addition to the buttons and menus in the user interface, Notify allows you to perform certain actions by pressing combinations of keys (keyboard shortcuts).

To open a certain window, from the main screen:

- **Alt + V:** Variables.
- **Alt + A:** Alarms.
- **Alt + C:** Contacts.
- **Alt + N:** Notification policies.
- **Alt + Q:** Queries.
- **Alt + W:** Actions.
- **Alt + H:** Message history.
- **Alt + L:** Log files.
- **Alt + S:** Main settings.

Editing:

- **Ctrl + E:** Switch to edit mode.
- **Ctrl + S:** Save changes.
- **Esc:** Cancel edit mode or close the current window.
- **F2:** Rename / Edit a field in the list.
- **Shift + F2:** Show edit window for a selected list item.
- **Ins:** Add an item to the list.
- **Del:** Remove items.
- **Ctrl + A:** Select all.
- **Ctrl + C:** Copy.
- **Ctrl + V:** Paste.
- **Ctrl + D:** Duplicate (equivalent to Copy & Paste).
- **Ctrl + Z:** Undo the last change, if available.
- **F9:** Select an item from value list, where available.

Other generic actions:

- **Ctrl + F:** Find or search.
- **Ctrl + L:** Login / User ID.
- **Ctrl + Shift + L:** Logout / Close session.
- **F1:** Help.
- **F7:** Quick Search / Filter.
- **F10:** Open the graph of a selected variable or alarm.
- **F12:** About...

23 FAQ / Troubleshooting

▪ **What functions does the DEMO version offer?**

After the first installation, and without the need to register or purchase a license, Notify starts in DEMO mode. Features and limitations of this mode:

- You need to accept the DEMO warning window for the application to start. This window closes automatically after 60 seconds of no interaction.
- The data historian and message notification systems are closed after 30 minutes of activity.
- The whole application closes after 2 hours but can be executed again manually.
- Within that time, the application is fully functional with all its available modules, with a limit of 2000 variables or alarms.
- You can use the DEMO mode to create or test any configuration of the project on any PC, being able to later export this configuration to a final installation that has a license.

▪ **What functions does the FREE license offer?**

When activating the *Free* license, without the need for registration, the application will stop working in DEMO mode, so it will not be closed after two hours. It allows the use of up to 5 variables / alarms, from a single data source, and only one email contact. It does not enable the use of SMS, queries, actions, writings, or data historian.

▪ **When an alarm is activated, no notification is received.**

The versatility of Notify when configuring the alarms and notifications system, makes it possible to forget activating something. The simplest way is to consult the main log file, find the alarm activation time and see what information the application is giving you. It must indicate which item is disabled so that it does not send the notification. In any case, you should check these points:

- ✓ The notification system and message type is enabled, configured, and working correctly, to send or receive messages as appropriate. Please send a test message if you want to verify it.
- ✓ If you have configured a daily or monthly limit for sending messages, verify that this value has not been reached.
- ✓ The type of message is enabled at a general level in "*Settings > Alarms > Activation Messages*" or "*Deactivation Messages*", as appropriate.
- ✓ The alarm meets the criterion of "*Limit priority to >= ...*"
- ✓ Verify if you have checked the box "*Do not send if there is another active alarm with higher priority...*", or there are dependencies between alarms, and the other alarm is active.
- ✓ There is a template configured for the alarm message. If there is a blank text, it will not be sent.
- ✓ The alarm has the "*Send*" box checked in the activation or deactivation settings, as appropriate.
- ✓ The activation time is within the alarm control time.
- ✓ The alarm is assigned to the contact.
- ✓ The contact is enabled.
- ✓ The contact has configured the necessary field for the type of message: phone number for SMS, or email address.

- ✓ The contact is active at the time of activation, according to the timetable that has been configured.
 - ✓ If all the above is ok, check the message history for the result of that message delivery. If it appears as OK, check what is indicated in the following FAQ point. If it indicates ERROR, check the log files for more information.
 - ✓ If you still cannot find a specific cause, there may be a problem with your GSM operator or the email server.
- ***The SMS appears as sent in the application but is not received on phone.***

It should be checked whether the operator needs the destination phone number of the message to be encoded with an international prefix, and in what format. Go to the configuration of the GSM modem in the main *Settings* and try to modify the parameter "*Default international prefix*". The prefix may have to be written in the "+XX" format, in the "00XX" format, or blank. Since this depends on the network, try altering the format and checking if the problem is solved.

- ***How to restrict the time of a contact for sending SMS, but keep 24 hours for sending emails?***

You can duplicate the contact in the phonebook. In one of them, configure only the phone number and the timetable when you want to receive SMS. In the other, configure only the email and the full time. If you use the *Copy / Paste* option, the assigned alarms are kept.

- ***My Gmail account does not work with Notify, what happens?***

Gmail blocks access from external applications to Google by default. To work with Notify, you must allow access from "less secure" applications. For it:

- Go to your Google account.
- In the navigation panel on the left, click *Security*.
- Turn on "*Less secure app access*".

Please note that Notify uses the security required by Gmail and other email services in terms of access to SMTP / POP3 / IMAP, activating encryption and secure transfer through TLS / SSL as required. The point is that Google considers any third-party application as "less secure" by default.

- ***No data column appears in the Variables or Alarms window ...***

The configuration of visible columns and sizes is persistent, so when the restarts, the user's preferences are recovered. On some occasions, for various reasons, these settings can be lost or saved settings can mismatch current screen settings, causing no data to be seen on these panels. To solve this, click with the right mouse button on the column header, and select the option *Select columns > Default*. This restores the original state.

- ***Can the application be updated / upgraded?***

Yes, depending on the license you have purchased. By default, paid licenses allow application updates for a period of one year, thus obtaining improvements, new functionalities, or bug fixes. However, at the time of purchase you can request supplements with longer update periods, or even for life.

If your license has a limited update period, it will not be valid for versions published after that date but will continue to work without limitations with the latest version available until that date. If you wish to purchase an upgrade, contact Tecmatia or your distributor, as there is available a reduced price compared to the standard rate.

- ***How to update the version of the application?***

If you have an Internet connection, you can do it from inside the application. In the "About..." window you will find the "Update" button, which checks online the availability of new versions.

If you have the installer for the new version, or download it manually from the web, just run it, closing Notify first if it is running. Follow the steps, which will already have loaded the current data of your installation and finish the process. Run the application again and you will see that the license and all the previous settings are preserved.

If when updating the application manually, the license is detected as invalid because the new version is more recent than the maximum update period, you can reinstall the previous version or request an upgrade of the license.

- ***My license does not include some features that I need, or I exceed the maximum allowable items, do I need to purchase a new license?***

Yes, but you will only have to pay for the characteristics you need. At any time, you can contact Tecmatia or your distributor to request the extensions you need.

- ***The application does not seem to work with my GSM modem ...***

Although the AT command interface is intended to be a standard, Notify implements the possibility of making adjustments in the set or syntax of the AT commands to be used, through the concept of a "driver" that you can select in the configuration of the GSM modem. By default, the most standard is "Generic GSM modem". However, some manufacturers provide AT commands with some discrepancy from the standard, which can cause communication errors in Notify. As we find these discrepancies, we add new drivers to the list with the necessary adjustments. If, despite this, you have a modem model that does not work properly with any of these drivers, please contact us and we will try to locate and correct this discrepancy in AT commands to make it compatible.

- ***What are the advantages / disadvantages of sending SMS through an online gateway compared to the GSM modem?***

They are subjective factors, but we could list some advantages:

- Ease of installation, as it does not require a physical modem with its wiring, antenna, power supply, etc., which is sometimes difficult to place, especially with low GSM coverage.
- The cost per SMS is usually lower than that of traditional telephony operators, although it will depend on the specific contractual conditions. Also, the initial cost of implementation is also lower since it is not necessary to purchase the GSM modem.
- The sending speed is higher, although it is not usually a determining factor in facilities that send few SMS.

- The reliability can be considered comparable or better. Most online services offer availability statistics of the order of 99.99%.
- The same account can be shared between multiple instances of Notify.
- In addition to Notify's internal message history, these services usually provide a web administration panel with additional history and statistics.

Regarding the disadvantages:

- It requires a reliable Internet connection.
- It is an additional provider, compared to the usual telephone provider.
- In the case of prepayment, it is necessary to pay attention to the remaining SMS balance, and carry out periodic recharges, although some gateways offer post-paid mode.
- It is not bi-directional. Although some gateways offer the option of renting a virtual number to allow incoming messages, now Notify does not implement this functionality due to the difficulty of homogenizing the different services, and only allows the sending of SMS. This implies that it cannot be used for Queries, Actions and Writings.

- ***The InfluxDB icon looks orange...***

There is an issue with the data historian in InfluxDB, but it is not an error that prevents execution. It may be due to incorrect measurement or tag settings, or database unavailability. Check the log for more details. Once the incident has been located and solved, you can recognize it by clicking on the icon. It will turn green again if everything is ok.

- ***I have an issue; I need help or a new feature ...***

Do not hesitate to contact Tecmatia or your authorized distributor for technical support, or let us to study the implementation of any new feature.



@ 2021, TECMATIA SL

Industrial Park PPI-7, Parcel 1, Ground Floor, Office A

36475 - O Porriño (Pontevedra) - Spain

Tel: (+34) 986 11 63 18

E-mail: info@tecmatia.com

Web: www.tecmatia.com