Administrator's guide

AutoCode for XProtect Corporate

General version:
2.8.10
Modules' versions:
xp.analyticsserver-1.4.0.25
xp.lpr-1.8.17.41
xp.configurator-2.3.4.1
xp.gateway-1.10.3.54
xp.viewer-1.4.2.15

vit
raising technologies™

© Video Internet Technologies Ltd., 2016
www.vitcompany.com/en

CONTENTS

1 introduction 4
- 1.1 Products Overview . . . 4
- 1.2 Manual Overview . . . 6

2 before starting 7
- 2.1 System Requirements . . . 7
  - 2.1.1 CPU . . . 7
  - 2.1.2 RAM . . . 8
  - 2.1.3 Network . . . 9
  - 2.1.4 OS . . . 9
  - 2.1.5 Surveillance System . . . 9
- 2.2 Preparing the XProtect System . . . 9
  - 2.2.1 Connecting to the Videostream . . . 10
  - 2.2.2 Registering Analytics Events . . . 16

3 installing modules 19

4 licensing 22
- 4.1 General Information . . . 22
- 4.2 Protection Tools . . . 23
- 4.3 License Activation and Update (Sentinel RUS Utility) . . . 23
- 4.4 License Control (Admin Control Center Application) . . . 26
  - 4.4.1 Sentinel Keys tab . . . 27
  - 4.4.2 Products tab . . . 27
  - 4.4.3 Features tab . . . 28

5 recognition settings 29
- 5.1 General Information . . . 29
- 5.2 Video Source Servers . . . 29
  - 5.2.1 Creation . . . 30
  - 5.2.2 Deleting . . . 30
  - 5.2.3 Configuration . . . 31
- 5.3 Target Servers . . . 31
  - 5.3.1 Creating . . . 31
  - 5.3.2 Deleting . . . 32
  - 5.3.3 Configuring . . . 33
- 5.4 Recognition Servers . . . 33
  - 5.4.1 Creating and Configuring . . . 33
  - 5.4.2 Deleting . . . 34
- 5.5 Channels . . . 35
  - 5.5.1 Creating . . . 35
  - 5.5.2 Deleting . . . 35

5.5.3 Configuring . . . 36
5.6 Templates . . . 47
5.7 Configuring Analytics Server . . . 49
5.7.1 Launch and Shutdown . . . 49
5.7.2 Connecting to XProtect . . . 50
5.7.3 Receiving Events from Third-Party Sources . . . 51
5.7.4 Monitoring Channels . . . 53
5.7.5 Configuring Logging Parameters . . . 54

6 configuring the viewer module 56

7 gateway module operation 60
7.1 Module overview . . . 60
7.2 Scripts . . . 60
7.3 Use case 1: Calculating time spent at parking . . . 62
7.3.1 Importing scripts . . . 63
7.3.2 Configuring scripts . . . 65
7.3.3 Determining special events . . . 66
7.3.4 Creating a property list . . . 66
7.3.5 Processing results . . . 69
7.4 Use case 2: Searching for recognized license plates in the list . . . 70
7.4.1 Local list . . . 70
7.4.2 External database . . . 77

a troubleshooting 80
a.1 Introduction . . . 80
a.2 Recognition sub-system settings: checklist . . . 82
a.3 License: checklist . . . 83

b messages 84
b.1 Events . . . 84
b.2 Alarms . . . 85
b.3 List of AutoCode XProtect analytics events . . . 86

1 INTRODUCTION

products overview

AutoCode XProtect (VIT) is a group of software modules for **XProtect** video management systems (Milestone Systems). The main purpose of **AutoCode XProtect** is to add video analysis functionality (namely, license plate recognition) to either a projected version of CCTV system or an already existing one. The modules are commonly applied in following use cases:

- highway traffic monitoring;
- automated access control systems development;
- border control;
- traffic monitoring at security checkpoints and secured premises such as parking lots, gas stations, repair shops, logistics centers, hotels, etc.

AutoCode XProtect modules (Fig. 1):

1. LPR is an **Analytic Server** module plugin. Main feature: software processing of a videostream for license plates detection and recognition. This module is capable of the following:
    - recognition of both single- and double-line license plates from over 50 countries;
    - recognition of up to 20 plates in the same frame;
    - maximum speed of the vehicle suitable for license plates recognition is 220 kmph (under proper LPR camera settings);
    - frame processing speed is within 10-100 ms depending on its resolution;
    - determining the direction of vehicle movement;
    - recognition fine tuning: license plate template selection, control over quality and parameters of resulting data.
2. Analytic Server is a Windows OS service in charge of license plate recognition process including sending the configuration data from **XProtect Management Client** to **LPR** plugin. The service manager allows activating/deactivating the **LPR** module, monitoring the processing of all incoming videostreams, creating system log entries as well as tracking module operation errors. Both **Analytic Server** and **LPR** modules are to be installed on each server devoted to video analytics.
3. Configurator is a plugin, necessary for license plates recognition configuration from within the **XProtect Management Client** interface. This module includes a tool for automatic validation of selected preferences.
4. Gateway is an **XProtect Event Server** plugin, set within **XProtect Management Client** user interface. This module is capable of the following:
    - filtering specific events generated within **XProtect** system;

- creating event processing logic using scripts or handlers (either provided with the product or created separately);
- working with external databases;
- creating unique events and sending them to the **XProtect** event server;
- working with third-party applications on SOAP level.

5. **Viewer** is a plugin for **XProtect Smart Client** created to display analytics events (including recognition results) in client application.

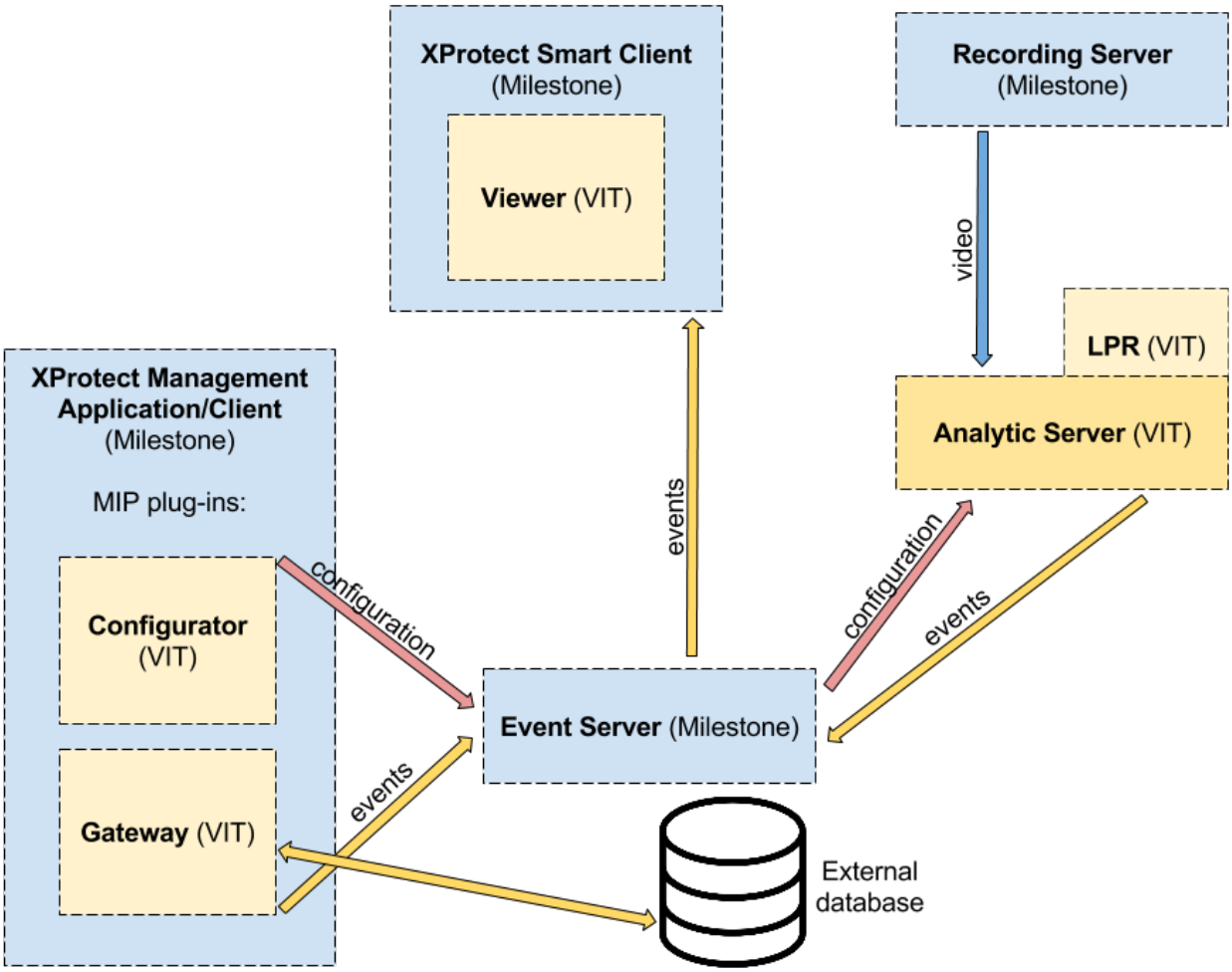


Figure 1: Communication between **AutoCode XProtect** modules and **XProtect** system components

Depending on the speed of vehicles in surveillance area, **LPR** module is provided in one of two possible modes — the parking mode and the freeflow mode. (see Table 1)

Table 1: Correlation between **LPR** module operation mode and the vehicle speed in surveillance zone

Mode	Video processing speed, fps	Speed of vehicles in surveillance zone, kmph
Parking	6 or less	20 or less
Freeflow	determined by framerate, licensing conditions and CPU capacity	220 or less

manual overview

This manual is designed for integrators, system administrators and **AutoCode XProtect** users with administrator rights. The manual is designed according to various tasks to be completed during different stages of working with **AutoCode XProtect**:

- before starting section contains the key hardware and software requirements and recommendations as well as instructions on setting up the **XProtect** system to enable the recognition sub-system.
- installing modules section contains instructions for installing the **AutoCode XProtect** modules.
- licensing section provides detailed instructions for license activating and updating process.
- recognition settings section provides:
  - instructions for setting up recognition using the **Configurator** module;
  - instructions for controlling and monitoring the recognition using the **Analytic Server** service;
  - checklists for primary troubleshooting in case any issues arise.
- configuring the viewer module section provides step-by-step instructions for setting up the recognition events to be displayed in **XProtect Smart Client** UI.
- gateway module operation section demonstrates the capabilities of the **Gateway** module using two examples:
  - calculating the time the vehicle spent on the parking lot;
  - searching for recognized license plates in external databases or local lists.

Legend:

> Important

> Additional information or example

2 BEFORE STARTING

system requirements

When the installation and configuration process for **AutoCode XProtect** modules is described in this guide, it is implied that the user has taken into account the **XProtect** system hardware and software platform requirements. Further sub-sections contain recommendations that, if taken into account, allow for providing a strong basis for the operation of license plate recognition sub-system within **XProtect** system. However, it is worth to keep in mind that neither the scrupulous selection of server configuration nor providing a stable Internet connection will have any effect if a low-quality videostream is fed for software processing. The best possible quality of a license plate image within the frame is the basic requirement that, if not met, makes receiving satisfactory results of recognition impossible.

> It is necessary to make sure that the video meets the requirements and recommendations provided in "License Plate Recognition Cameras Selection, Setup and Installation Guide" document.

Generally, on software and hardware platform preparation stage, a special attention should be paid to:

- selecting a CPU (for the recognition server) since it receives the most workload during the recognition process;
- calculating network throughput, sufficient to transmit the videostream without losing frames that are to be processed;
- compatibility of **AutoCode XProtect** modules versions with **XProtect** system and Windows OS.

CPU

The performance of recognition sub-system directly depends on CPU capacity. This performance is measured in total amount of frames processed per second on all recognition channels.
When the **LPR** module is working in freeflow mode, this index should equal the framerate of the videostream. Keep in mind that when working in parking mode, it is impossible to process more than 6 frames per second.
When the capacity of the CPU is not sufficient for selected recognition settings, a decimation occurs in the set of frames causing losses of potentially important information. This leads to a decrease in recognition quality and percentage.
Table 2 contains the number of frames per second that can be processed by recognition module under specific computer configuration, processor load and image resolution. This data is a result of performance testing under the following conditions:

- The videostream is transferred with minimum level of compression (Xvid or MJPEG formats are used) meaning that compression workload is not included into calculations.
- The CPU is loaded with recognition only.
- Default configurations with CPUs of various price categories and performance rates (from lowest to highest) were selected for testing.

Default configurations with CPUs of various price categories and performance rates (from lowest to highest) were selected for testing.

Table 2: Comparing the operating rate of **LPR** module under 60% CPU workload (fps)

Computer configuration	Image resolution			
	720*576	1360*512	1360*896	2336*1753
Intel Core i5-2500 3.3Ghz 2Gb RAM	96	51	28	7
Intel Xeon DP Quad-Core E5620 8Gb RAM	135	70	40	11
Intel Core i7-2600 3.4Ghz 8Gb RAM	165	76	49	15
Intel Core i7-3930K 3.2Ghz 16Gb RAM	277	144	82	24
2* Intel Xeon X5650 2.66Ghz 12Gb RAM	426	224	135	40

When the videostream is transferred for processing or displaying in the client application, it is being decompressed. After that, the whole videostream is decoded disregarding whether the recognition sub-system is working in either freeflow or parking mode. Therefore, even if the module runs recognition at 6 fps, the whole videostream received from the camera will be decoded (e.g. 25 fps).
A portion of server resources may be used for additional tasks such as video archive recording, interaction with the database, anti-virus protection, etc. With this in mind, it is recommended to pick a CPU that, with the recognition sub-system actively running, would have about 60% workload. When picking the CPU, one can use the online calculator available on **AutoCode XProtect** developer's website.
If the selected configuration matches one of those provided in table 2, it is possible to determine approximate available number of recognition channels by dividing total number of frames taken from the table by framerate of selected recognition mode.
For example, under 1360*512 resolution, Intel Xeon DP Quad-Core E5620 8Gb RAM configuration and 60% CPU workload, **AutoCode XProtect** will be able to process video from:

- 11 cameras in *parking* mode ($70/6 \approx 11$);
- 2 cameras in *freeflow* mode if their framerate is 25 fps ($70/25 \approx 2$).

> Example of default hardware configuration:
> Intel Core i5-2500T (2.3Ghz), 2Gb RAM. Camera: Axis P1344. Codec: H.264. Fps: 20. Resolution: 1280*800. Channels (with decoding): 3.

RAM

Adding recognition channels requires respective adjustments to the amount of RAM. One recognition process (one channel) uses up to 200 Mb of RAM. In case the recognition process on one channel is divided into multiple *threads*, RAM usage of this process would increase accordingly.

> The term **recognition thread** is defined in the description of the **Channel** system object (see the Additional Settings tab description).

Network

If the recording server and the license plate recognition sub-system (the **LPR** module) are located on different computers, it is important to setup a stable connection between them with sufficient network throughput. This helps avoid interruptions that cause important information being lost.

OS

AutoCode XProtect modules are compatible with following operating systems: Windows 8 Enterprise, Windows 8 Pro, Windows 8.1 Enterprise, Windows 8.1 Pro, Windows 7 Ultimate, Windows 7 Professional, Windows 7 Enterprise, Windows Server 2008 R2, Windows Server 2012 and Windows Server 2012 R2.

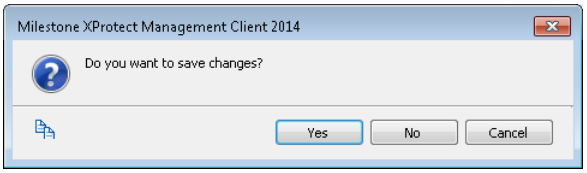
Surveillance System

The integration of **AutoCode XProtect** modules into **XProtect** system is realized using **MIP SDK 2014**. Therefore, the modules are compatible with the versions of the system that support plug-in integration: Enterprise, Corporate, Express, Expert, Professional.

preparing the xprotect system

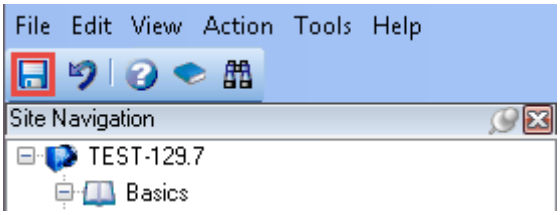
LPR sub-system can only be configured if **XProtect** system is configured according to the following requirements:

- The videostream source is connected. The procedure of importing the video from a file is described further in this section. Detailed instructions for connecting to hardware devices are provided in **XProtect** Administrator's Manual.
- The special analytics events necessary for receiving the results of recognition within **AutoCode XProtect** are configured.



> The dialog window above appears after any changes made in configuration. The buttons in the window provide control over the configuration process:
>
> - **Yes** — save the changes.
> - **No** — discard all changes made after the configuration was saved previously.
> - **Cancel** — close the dialog window to proceed with configuration delaying saving.

Additionally, the configuration can be saved by clicking the **Save** button on the toolbar next to main menu of the application:



Connecting to the Videostream

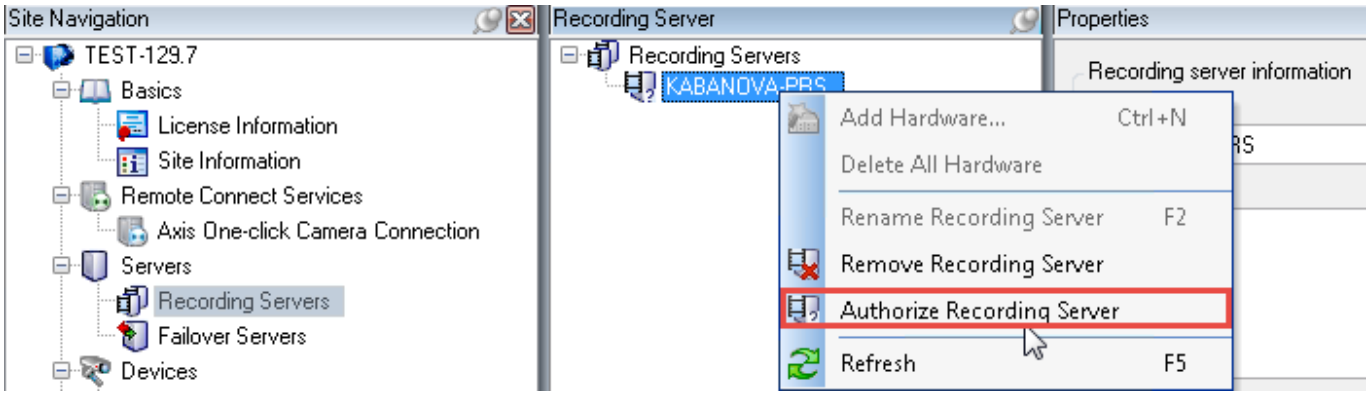
In order to test LPR sub-system or use it on a regular basis, the source of the video must be connected. The process of adding cameras and other hardware devices, such as video encoders, to **XProtect** is described in this system administrator's manual. The following instructions are describing the process of importing the video as the data source.

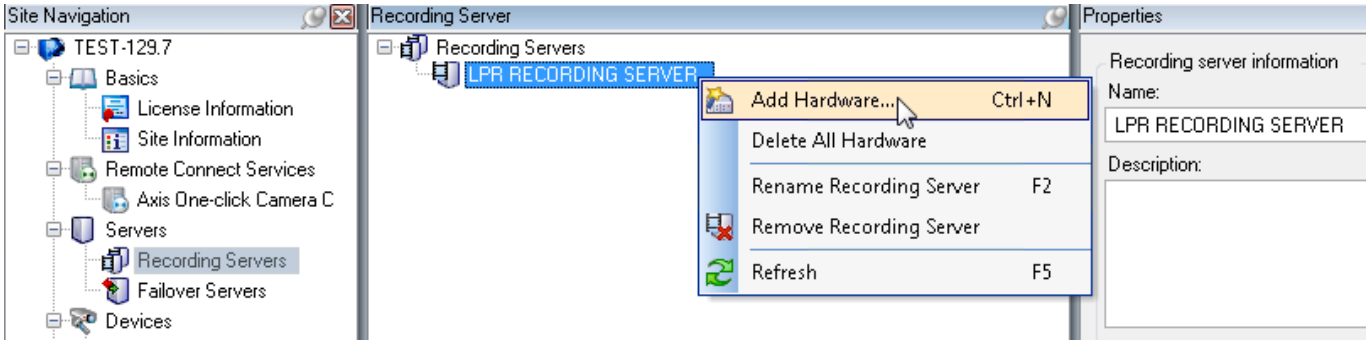
Step 1. Adding and configuring devices

Recording server is a server that, within **XProtect** system, provides connection between IP-cameras and video encoders to record received audio and video stream. Using recording server, user can access archived data as well as live video. Recording server feeds video streams to be processed by **LPR** module. Recording server functionality is necessary to create cooperation between **XProtect** products combined within **Milestone Interconnect** — a single multi-site surveillance system.
To import a video file into **XProtect** system:

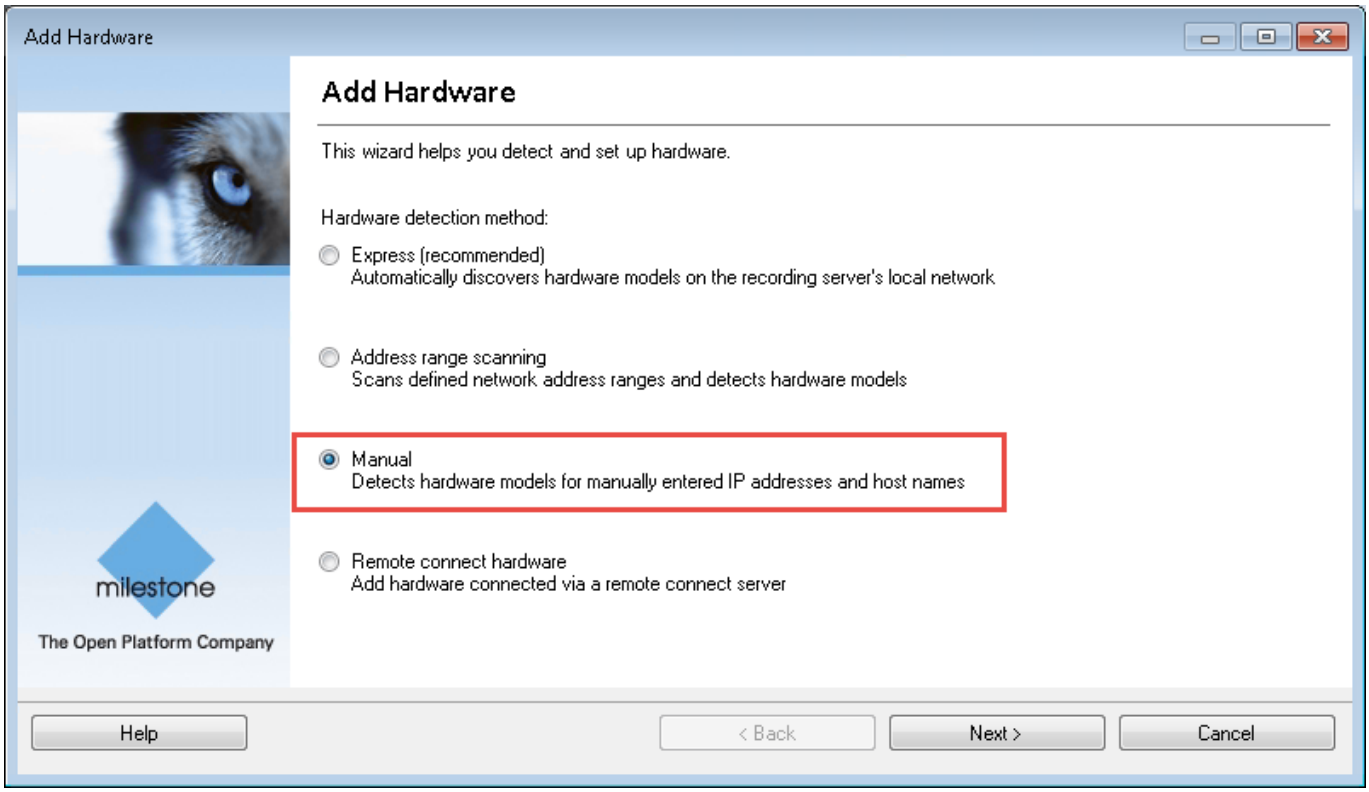
1. Select **Recording servers** object on **Management Client** object tree. The overview panel on the right will display all recording servers currently connected to host of the CCTV system and connected to **Management Server**.

2. Open context menu of an authorized recording server and select **Add Hardware**.

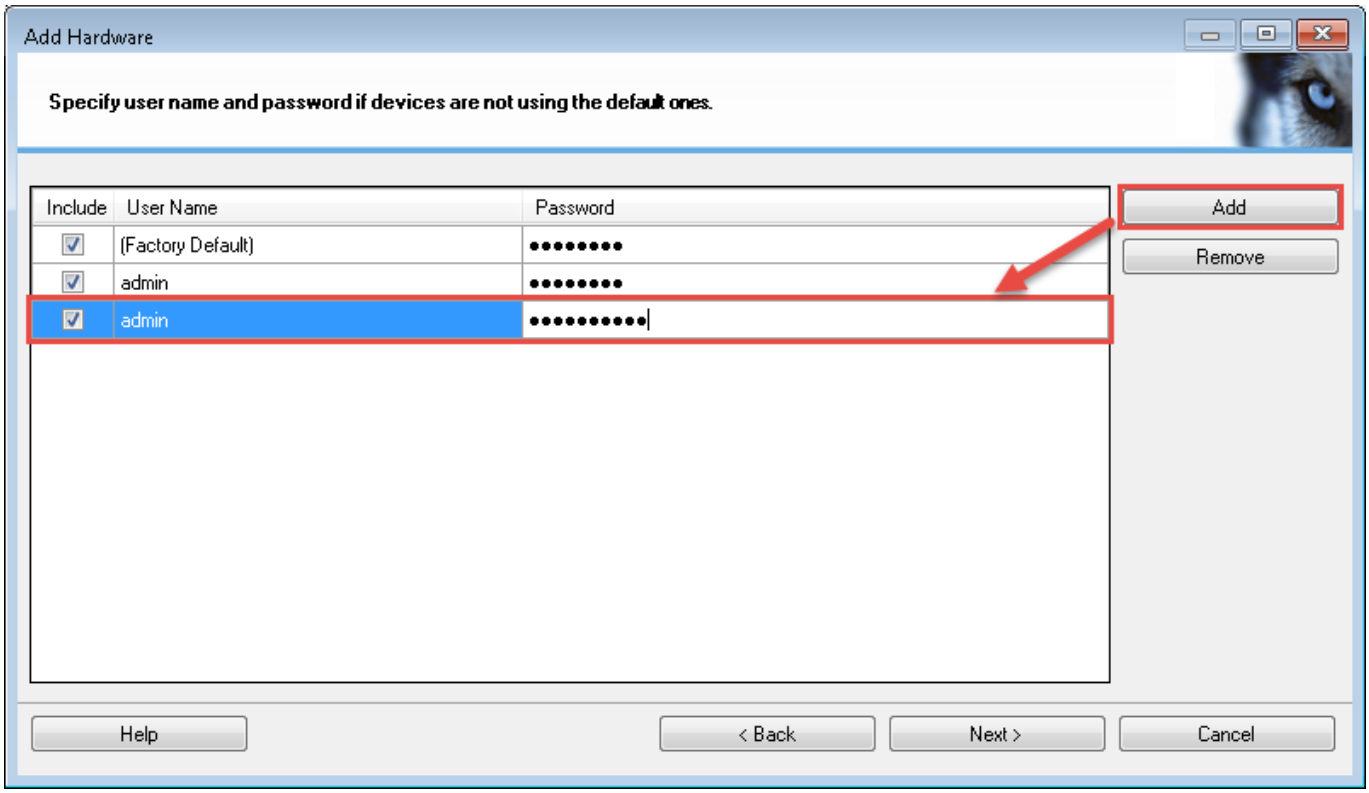




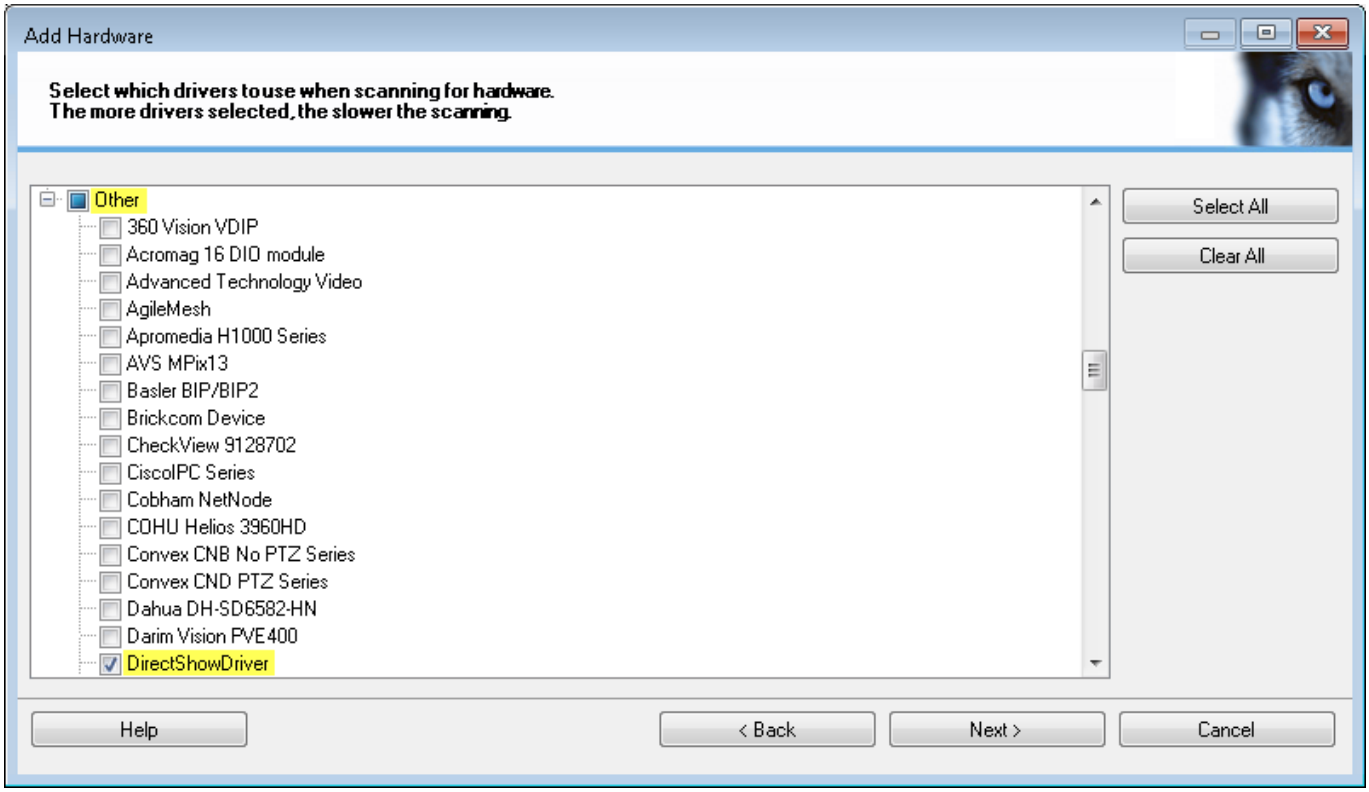
3. Select **Manual** in **Add Hardware** wizard window.



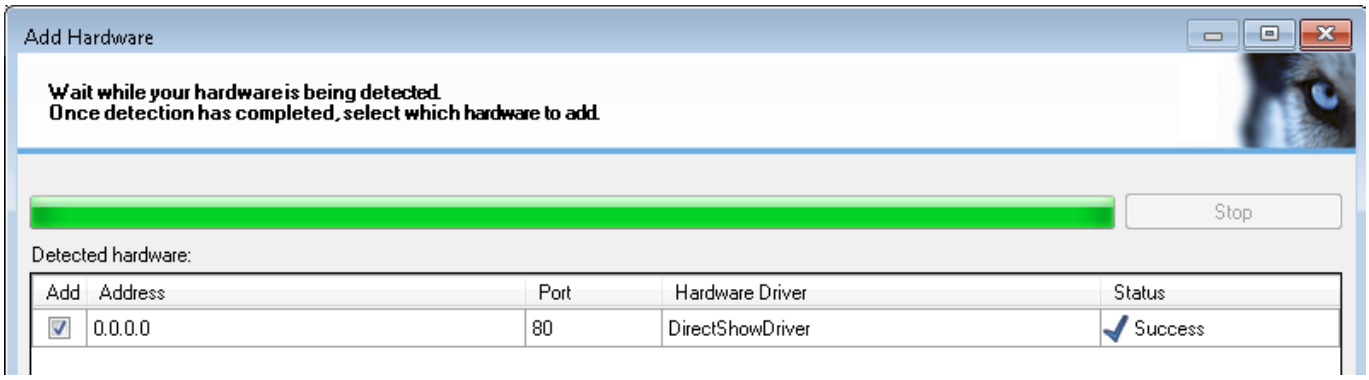
4. On the next screen of wizard window, click **Add** button to provide the following data for the new device (case sensitive): **User Name** —admin; **Password** —DirectShow. **Include** checkbox must be selected.



5. Select **DirectShowDriver** (**Other** list).



6. Specify parameters for connecting to a new hardware device: **Address** — 0.0.0.0; **Port** — 80; **Hardware Driver** — DirectShowDriver. Click **Next**. Check if the hardware was successfully recognized (**Success** status).



Name created hardware devices (Fig. 2, 3, 4, 5). Click **Next**.

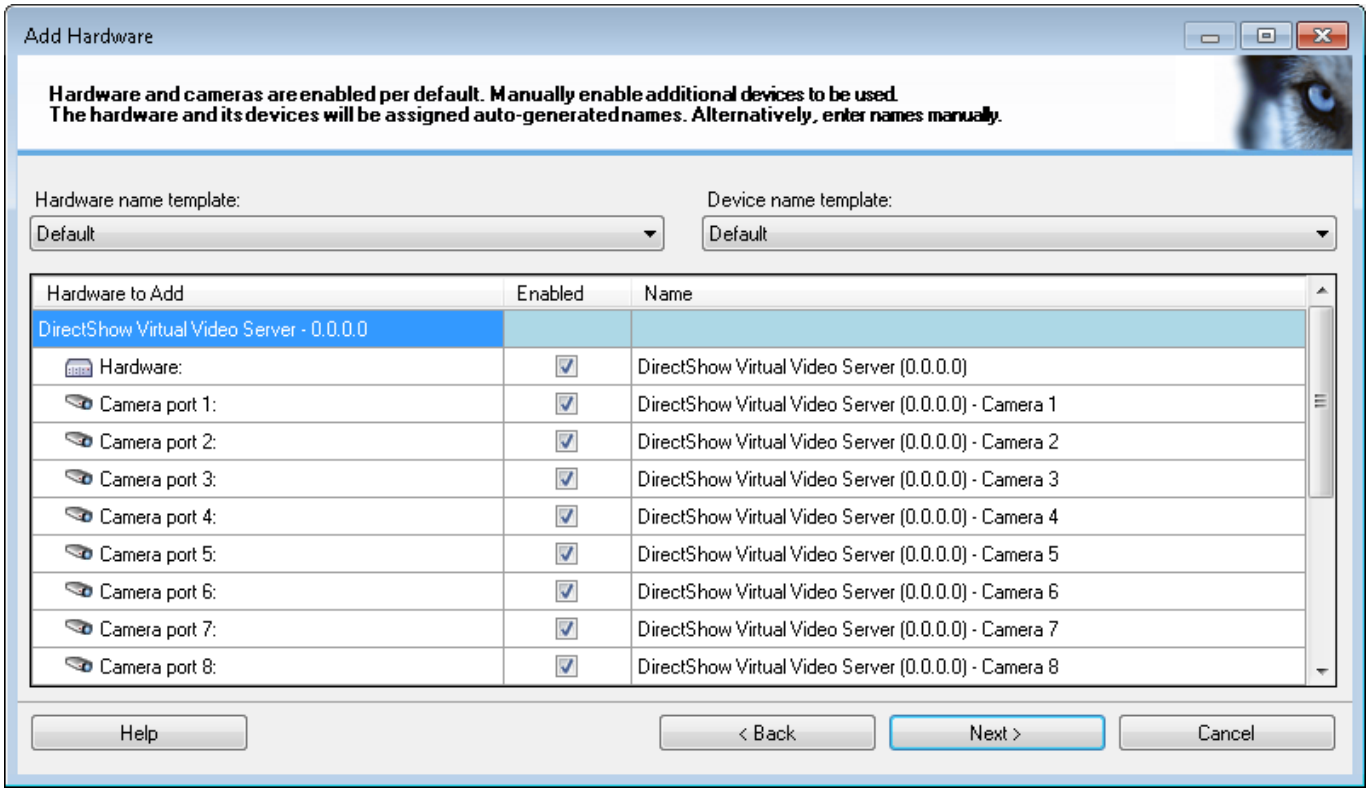


Figure 2: Default hardware names (created automatically)

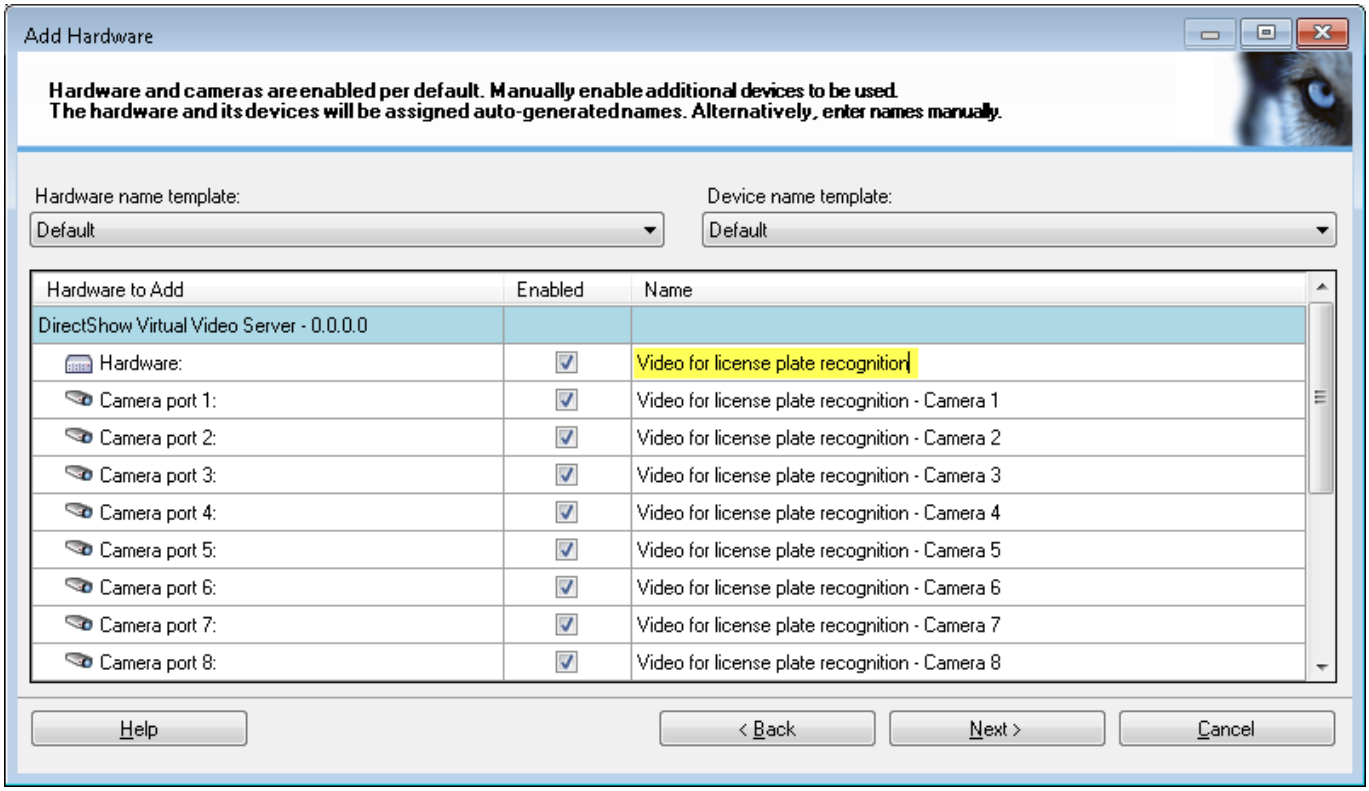


Figure 3: Editing names manually

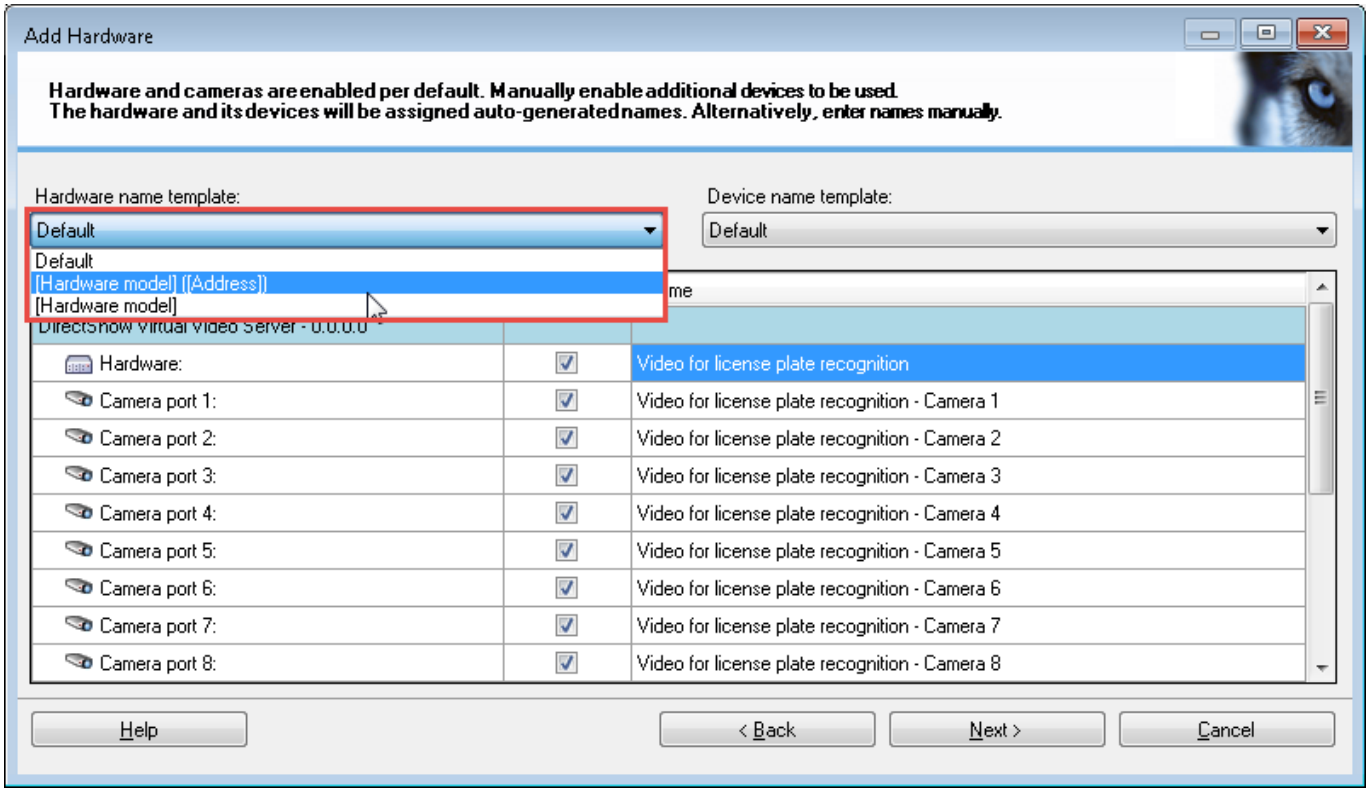


Figure 4: Selecting a hardware name template

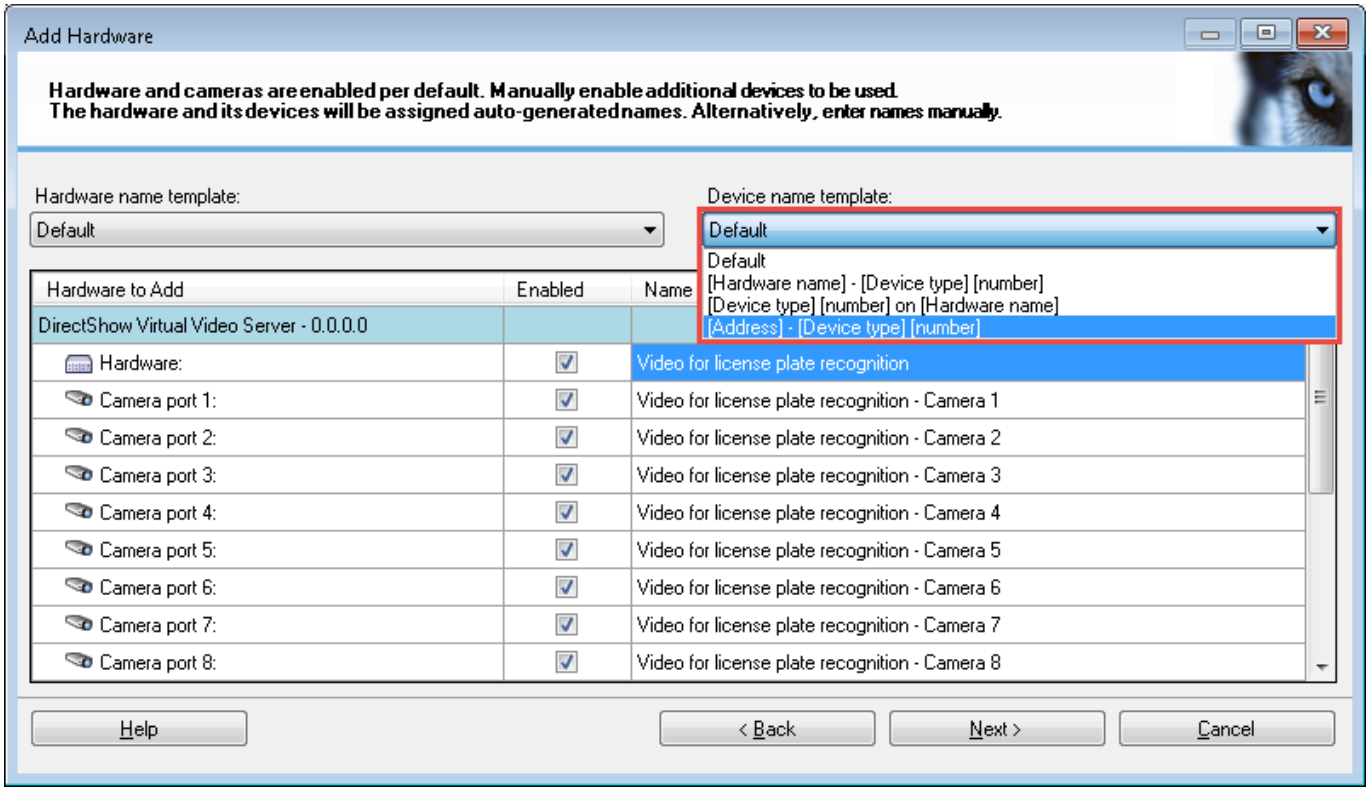
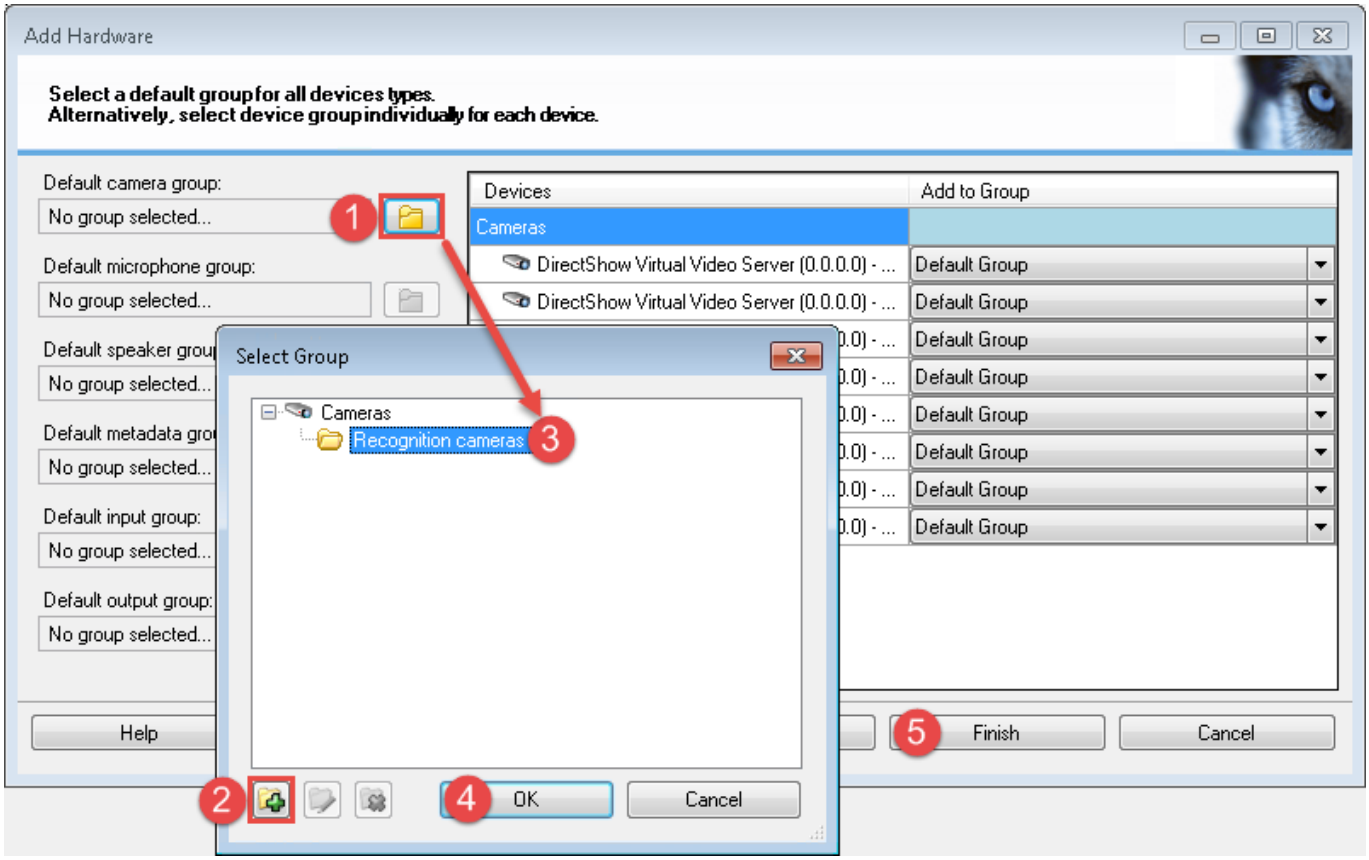


Figure 5: Selecting a device name template

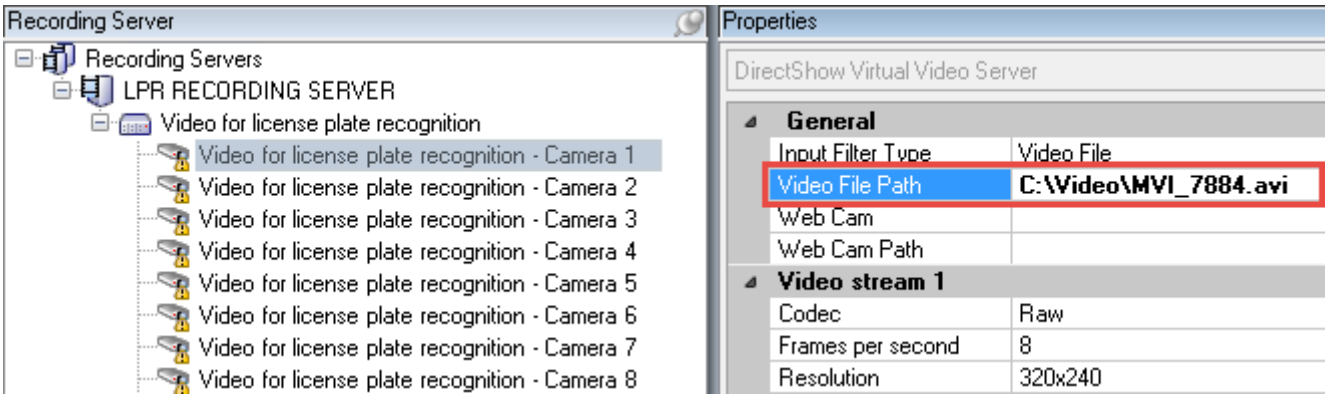
7. Create/select a camera group (e.g. **Recognition cameras**). Click **Finish**.



Step 2. Setting up a Camera object

The second step of connecting to a videostream is configuring one of **Camera** objects created on a new hardware device:

1. Select **Camera** object on the object tree. The overview panel on the right will display all objects of the kind that were created in the system.
2. Select an optional instance of **Camera** object. On the object properties panel on the right, switch to **Properties** tab.
3. Specify the file path.
4. Save the configuration.



Registering Analytics Events

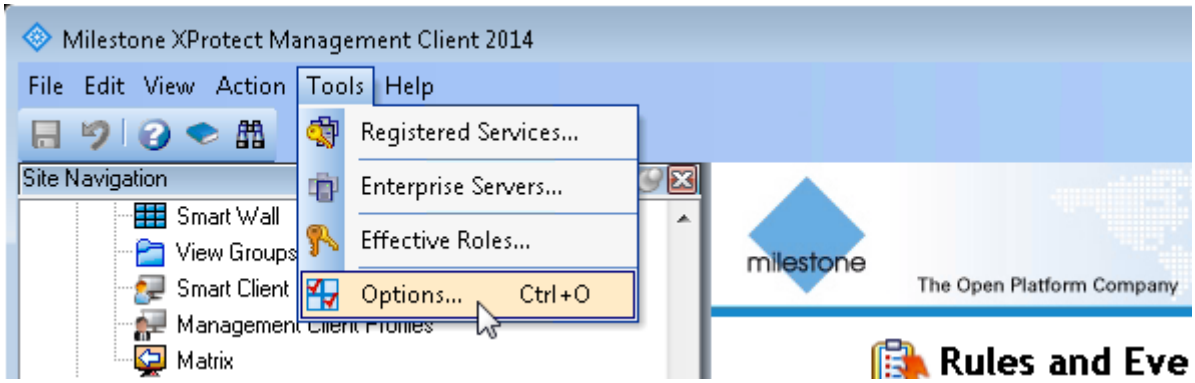
Analytics events are resulting from the operation of additional software products integrated into the **XProtect** system for intellectual processing of video data. This includes the **LPR** and **Analytic Server** as LPR sub-system within the set of **AutoCode XProtect** modules.
The following events are generated during the recognition:

- **Plate detected** — the fact of receiving the best possible recognition result. This event depends on **Send event while car in frame** setting of the **Channel** object:
  - If this setting is *deactivated*, the event is generated after the vehicle has left the frame (along with the **Car lost** event).
  - If this setting is *activated*, the event is generated when the vehicle enters the frame (along with the **Car detected** event).
- **Car detected** — the fact of receiving the first recognition result that registering that a vehicle entered the frame. It depends on **Result Delay (msec) setting** — the event is generated after the time specified in the setting.
- **Car lost** — the fact of the license plate leaving the recognition zone. Depends on **Check license plate overlap (sec)** setting. If the license plate does not reappear in the frame within the time period specified in the setting, it is considered to not be overlapped, so the vehicle has left the frame.

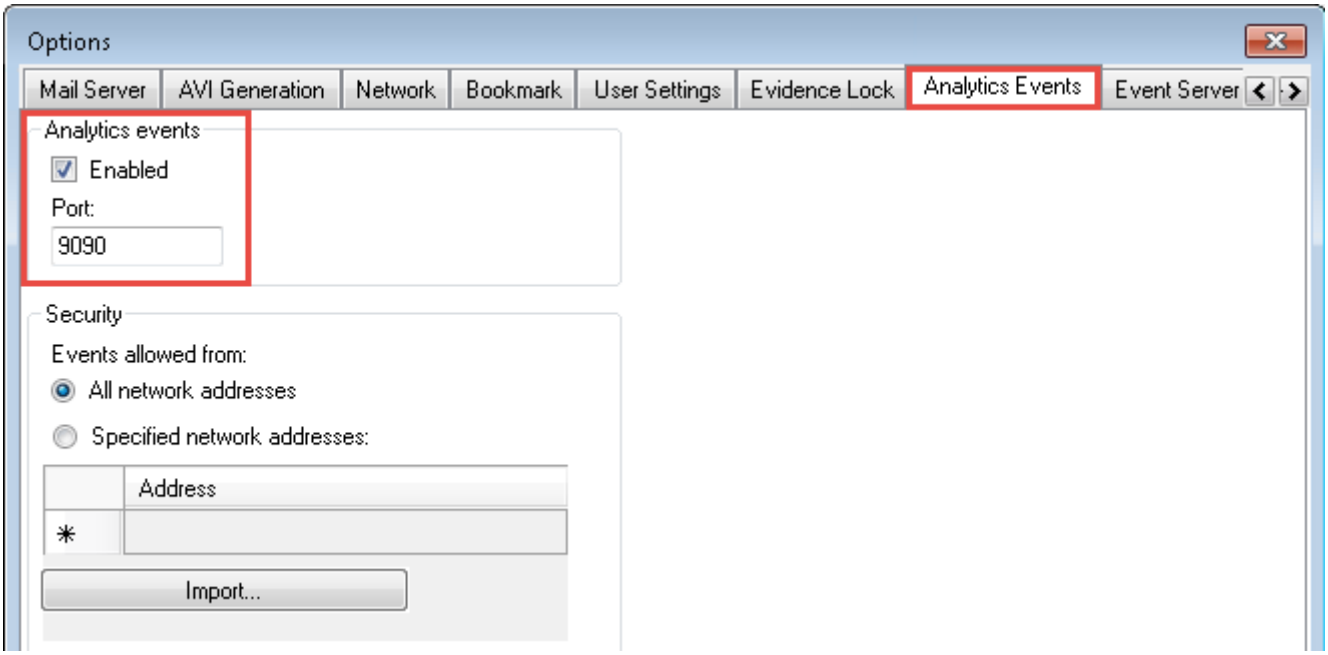
> **Plate added** and **Plate detected** events may be generated when working with **XProtect Smart Client** if the license plate is added/edited manually. Such events should only be registered in **Management Client** if the alarms are to be generated based on them.

To have the analytics events transferred from LPR sub-system to event server, they should be identified (marked for transfer). The following steps are necessary for this process:

1. Open **Tools** menu and select **Options**.

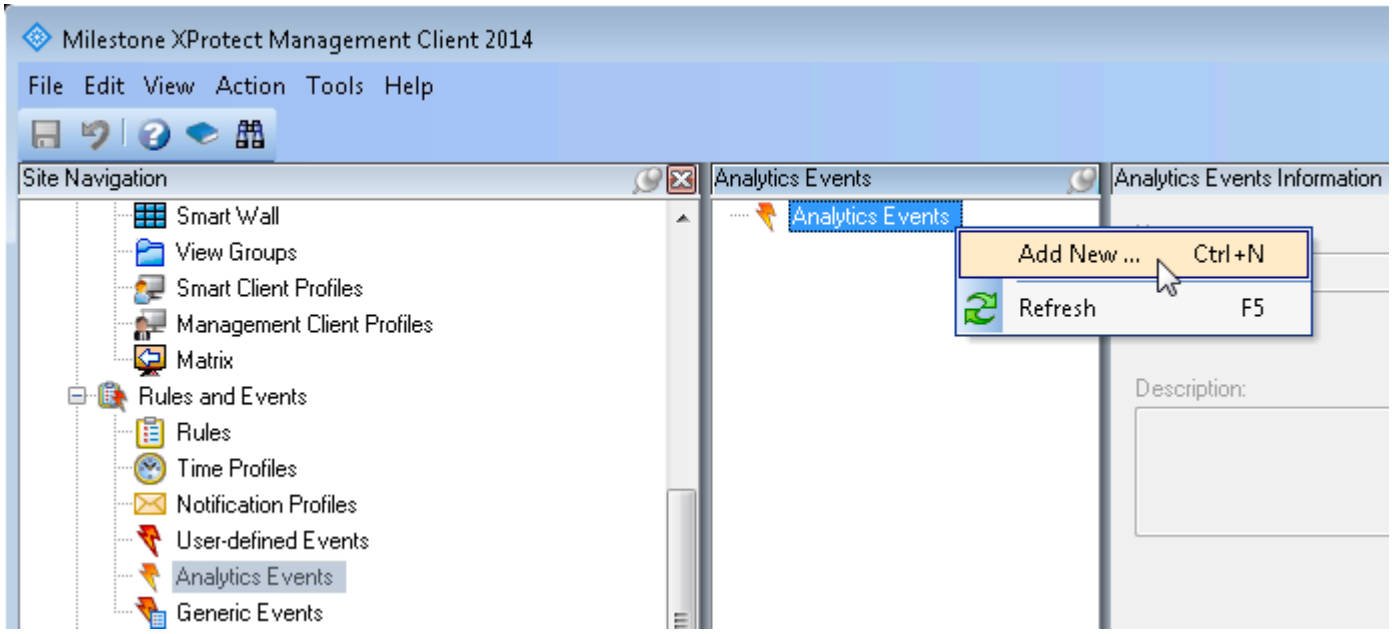


2. In the window that appears, switch to **Analytics Events** tab. This tab contains the settings overseeing usage of analytics events within **XProtect** system in general.
3. Make sure that the system is allowed to use analytics events (**Enabled** checkbox is selected).

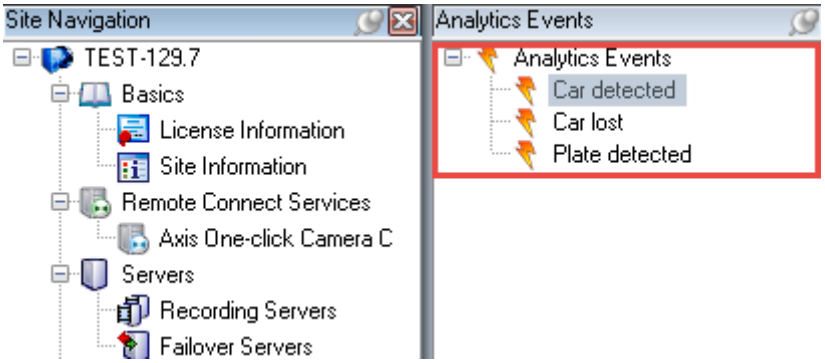


> Other settings on **Analytics Events** tab can be used to allow analytics events to be received from either all network addresses or a specified range of addresses. A detailed description of this tab may be found in **XProtect** Administrator's Manual.

4. Open the context menu of **Analytics Events** object (located under **Rules and Events** parent object on the object tree). On the overview panel located to the right select **Add New** option.



5. Repeat the aforementioned steps to create **Car lost** and **Plate detected** events.



When creating identifiers for events, a special attention must be paid to the register: upper case is to be used on the first letter of the first word while the lower case is to be used for remaining symbols. Both words are to be separated by a space.

3 INSTALLING MODULES

To have LPR sub-system function correctly, the **Analytic Server** module must be installed prior to installing the **LPR** module. The remaining **AutoCode XProtect** modules may be installed in arbitrary order.

1. Extract them.
2. Run the installation files as an administrator and perform all the steps necessary for the installation.
3. Reboot the computer after installation is finished.

Notes for installing **AutoCode XProtect** modules:

- **Configurator**. Installed on one computer:
  - with **Management Client** application pre-installed and **XProtect Management Client Plug-in** option selected on **Select Components** stage (installation wizard) (see example on Fig. 6).
  - with **XProtect event server** if a different host is used and with **XProtect Event Server Plug-in** option selected on **Select Components** stage (installation wizard).

  In cases when both **Management Client** and event server are located on the same computer, all components will be installed by default (**Full version** option).

  > On **Types of recognition channels** stage, the **Only LPR** is to be selected (Fig. 7). This must be done to allow for the recognition system (**LPR** and **Analytic Server** modules) to be configured.

- **Gateway**. Installed on one computer:
  - with **Management Client** application pre-installed and **XProtect Management Client Plug-in** option selected on **Select Components** stage (installation wizard) (see example on Fig. 6).
  - with **XProtect event server** if a different host is used and with **XProtect Event Server Plug-in** option selected on **Select Components** stage (installation wizard).

  Additionally, a license key with **Gateway** module license is to be installed on this computer. A path to either **XProtect Event Server** or **Management Client** is to be specified at one of installation wizard stages.

  In cases when both **Management Client** and event server are located on the same computer, all components will be installed by default (**Full version** option).

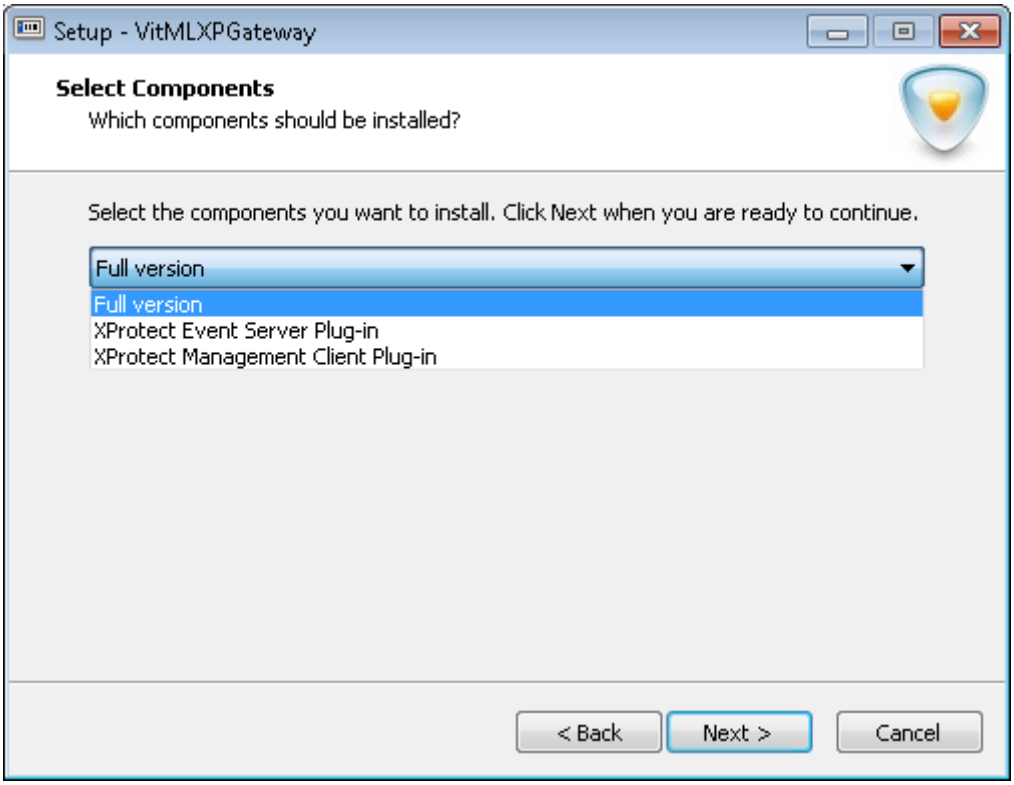


Figure 6

- **Analytic Server**. This module is to be installed on a computer used as the recognition server. The license keys (hardware or software, see the Protection Tools section) are to be connected to this computer.
- **LPR**. This module is to be installed on the computer that has **Analytic Server** module pre-installed.
- **Viewer**. This module is to be installed on the computer that has the **XProtect Smart Client** application pre-installed.

HDD space required to store **AutoCode XProtect** modules:

- **Configurator** —180 Mb. For **XProtect Corporate** integration with both event server and **Management Client** located on the same computer — 250 Mb.
- **Gateway** — 230 Mb. For **XProtect Corporate** integration with both event server and **Management Client** located on the same computer — 320 Mb.
- **Analytic Server** — 380 Mb.
- **LPR** — 1.7 Gb.
- **Viewer** — 145 Mb.

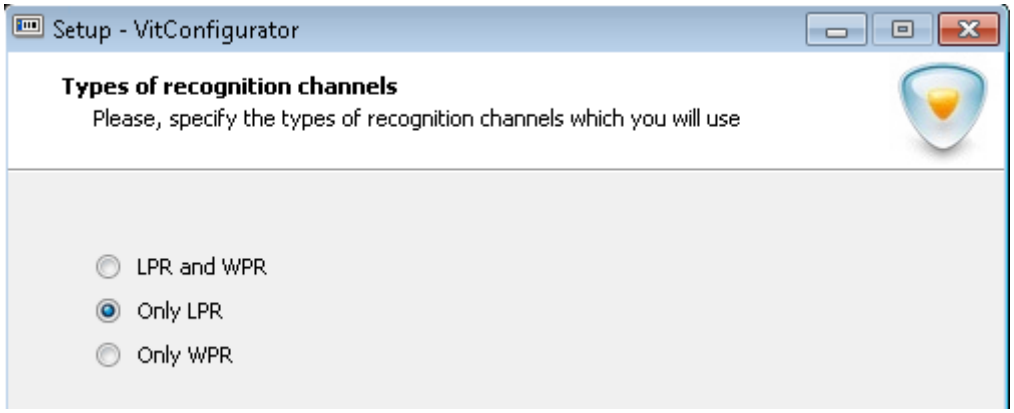


Figure 7: Installing **Configurator** module: selecting LPR analytics type

Additional software is provided within **AutoCode XProtect** installation packages ('_redist' folder), including:

- **ffdshow** codec used by **LPR**;
- **Sentinel Runtime** driver for license keys;
- **Sentinel RUS** remote license update manager;
- **Microsoft .NET Framework 4**;
- **IronPython 2.7.4** used by **Gateway**.

Aforementioned components are installed automatically.

> Demo versions of **AutoCode XProtect** do not work on virtual machines or under active RDP sessions. Using the modules on VMs is only possible after purchasing the hardware license.

4 LICENSING

general information

License protection of **AutoCode XProtect** modules is carried out using software and hardware products developed by SafeNet.
General licensing parameters include:

- **Vendor**. A unique identifier of the company that developed the protected product. **AutoCode XProtect** modules are protected using keys from EAOWT and AAOTB series (numbers equivalents are 107392 and 106763 respectively).
- **Firmware (License)**. A feature of the license key that determines the set and capabilities of product functionality available for operation. Varies based on the requirements of the client.
- **Feature**. An identifier for a licensed functionality unit of the product. The set of features determines the firmware of a license key.

> The software product components (accessed using a license key) are available for display and control using the **Sentinel Admin Control Center** application. This application is described in the License Control (Admin Control Center Application) section.

Should a specific component be missing from the firmware of a license key, a respective functionality will not be available to the user.
The following **AutoCode XProtect** software products are to be protected:

- **Analytic Server** module (Feature 471)
- **Gateway** module (Feature 441)
- **LPR** module (Feature 17 and 440)
- Number of countries, license plates from which are to be recognized (Features 4000-4256 with Feature 4000 being obligatory)
- Number of recognition channels and video processing speed (Features 1-390):
  - 'Slow' ('parking') — up to 6 frames per second. Used to recognize license plates on vehicles moving under 20 kmph, e.g. at parking lots.
  - 'Fast' ('freeflow') — defined by framerate of the camera used as well as the processor capacity. Used to recognize license plates on vehicles moving faster than 20 kmph (on highways).

> The term *recognition channels* is described in Channels section (see the description of Additional Settings tab).

protection tools

Operation of **AutoCode XProtect** modules depends on **Sentinel LDK** (**Sentinel Runtime**) license key driver. Version 6.65 or higher is to be used. **AutoCode XProtect** modules can function only if license keys are installed locally (see the table 3).

Table 3: Features of **Sentinel HASP** key based on its type

Key type	Description
HASP HL Key	Provided as a hardware token (USB-key). This key is physically installed onto a server with the protected software to be used. The key may be transferred without losing the license.
HASP SL Key	Provided as a service. It is attached to hardware of the computer. If the modules are transferred to another server, a new license should be purchased.

license activation and update (sentinel rus utility)

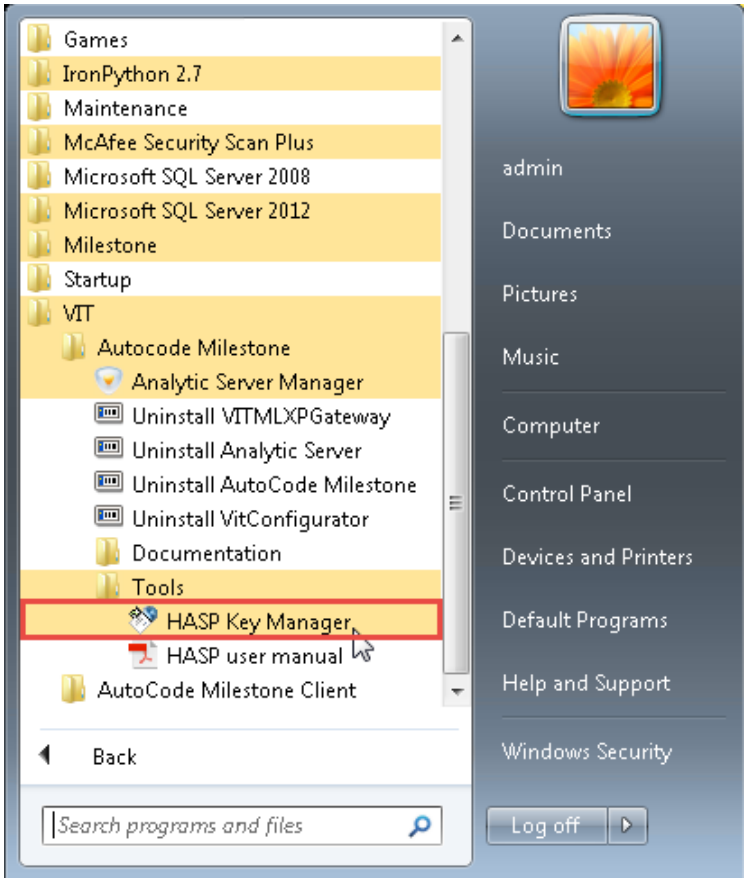
License update is to be carried out using **Sentinel RUS** (**Remote Update System**) utility. Name of the program file — **hasp_update.exe**.
Sentinel RUS license update manager will become available only after all necessary **AutoCode XProtect** modules are installed.

Step 1. Launching the Sentinel RUS utility

When using a hardware license key, it is necessary to make sure that:

- it is connected to the server;
- its driver (**Sentinel LDK**) is installed.

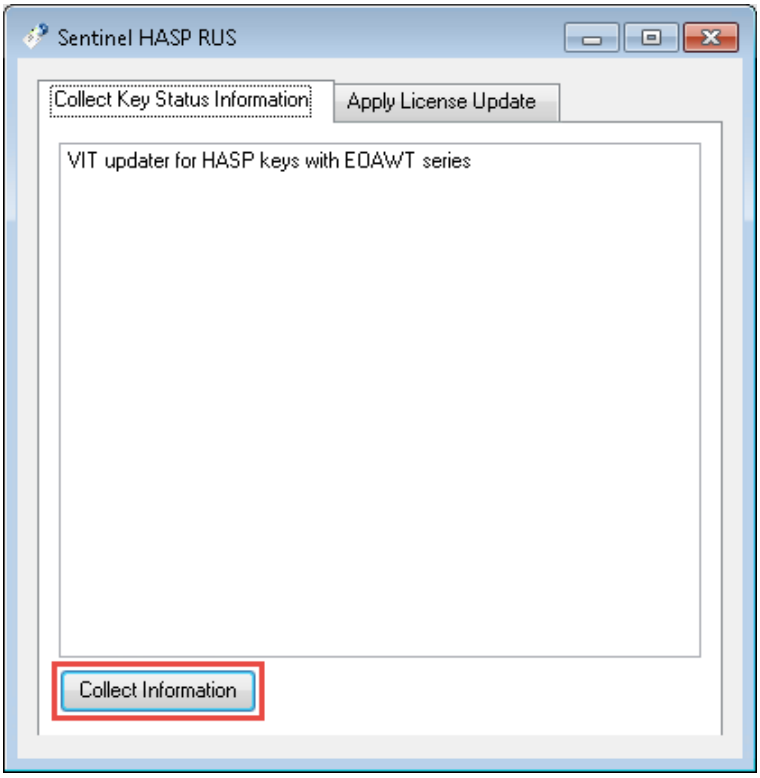
Sentinel RUS is provided within the installation package of every **AutoCode XProtect** module that requires protection (**Analytic Server**, **LPR**, **Gateway**).
The utility may be launched from the directory of one of the modules (by default, file path is C:\Program Files (x86)\VIT\[name of the module]) as well as from **Start** menu.



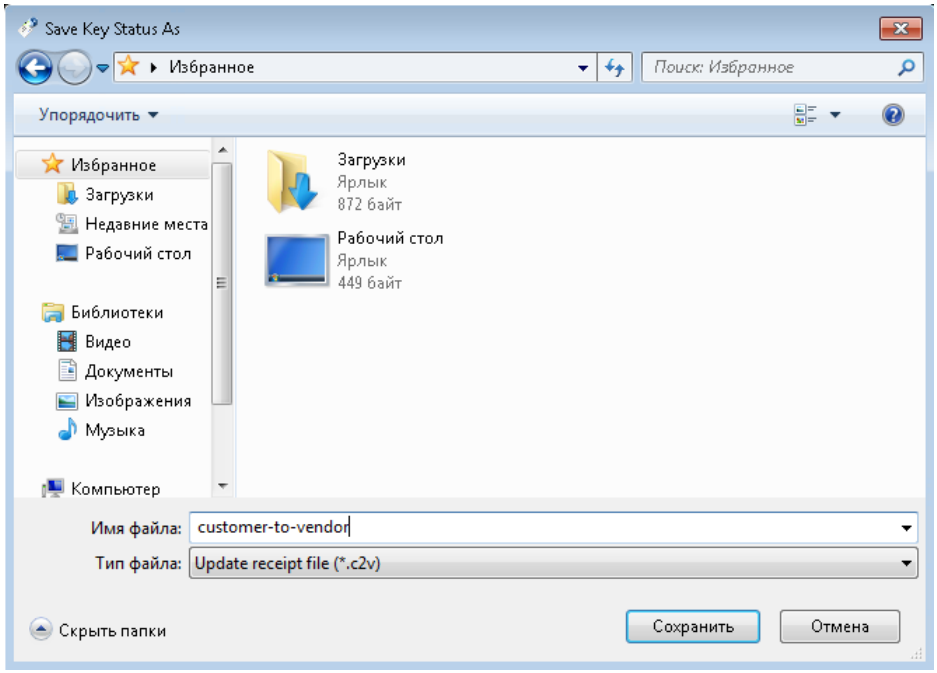
Step 2. Generating a c2v file for the manager

At this point, a c2v (customer-to-vendor) file reflecting current state of the license is to be generated and sent to the manager. The following steps are to be taken to do it:

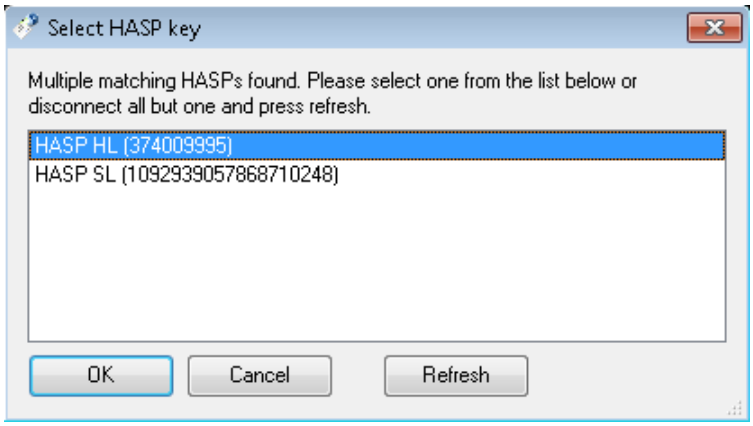
1. Click **Collect Information** button in the **Sentinel HASP RUS** window.



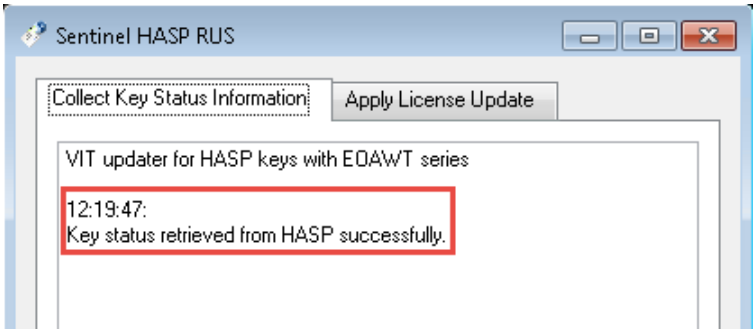
2. Specify the name for the generated c2v file and save the file to a specific directory.



If there are more than one license keys installed on a local computer, a proper key to save the state should be selected when generating a c2v file. To get details on license keys currently available on the server, use **Admin Control Server application** — see the License Control (Admin Control Center Application) sub-section.



3. Make sure the c2v file is created successfully and send it to manager.



A c2v file can also be generated using **Admin Control Server** application. However, in order to successfully update the license to use **AutoCode XProtect** modules, the file must be generated by **Sentinel RUS** utility.

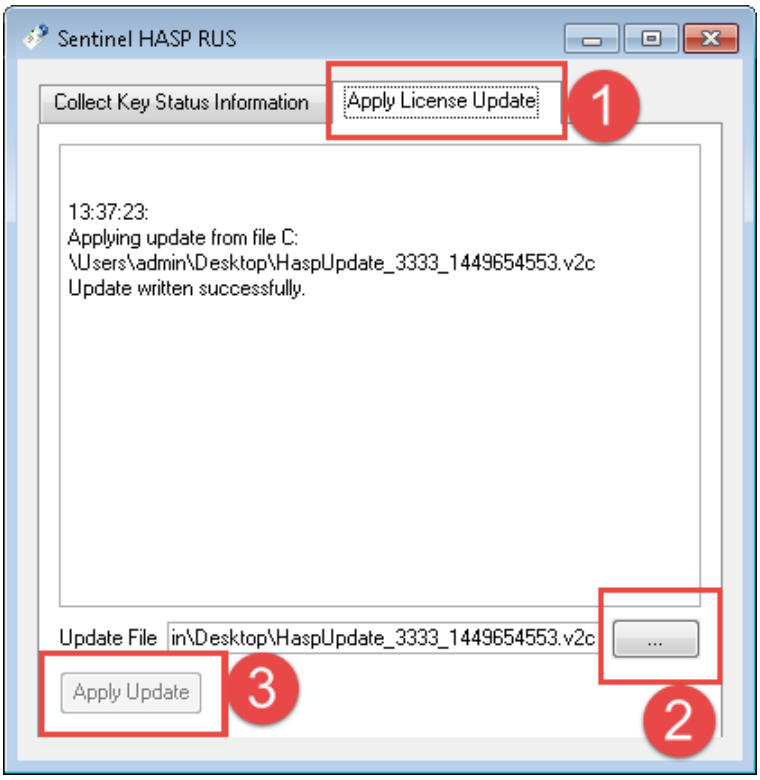
> On the other hand, it is possible to use a file generated by the **Admin Control Server** application (see Step 4).

Step 3. Receiving updated license (v2c file) from the manager

The manager should return a .v2c ('vendor-to-customer') file. This file is necessary for license activation/update.

Step 4. Applying received v2c file

Open **HASP Key Manager** again and go to **Apply License Update** tab (1). Specify the path to a directory (2) that holds the v2c file received from the manager, select the file and click **Apply Update** button (3).



license control (admin control center application)

Sentinel Admin Control Center application is used to control licenses on both local and network servers as well as to diagnose associated problems. This section contains a brief description of the application. A detailed description, including a description of tabs not mentioned in this manual, such as **Access Log** and **Configuration**, can be found in the **Help** section of the application.

> The application is to be opened using web-browser on a local computer with **Sentinel LDK License Manager** service and **hasplms.exe** process active. The URL is http://127.0.0.1:1947/_int_/devices.html.

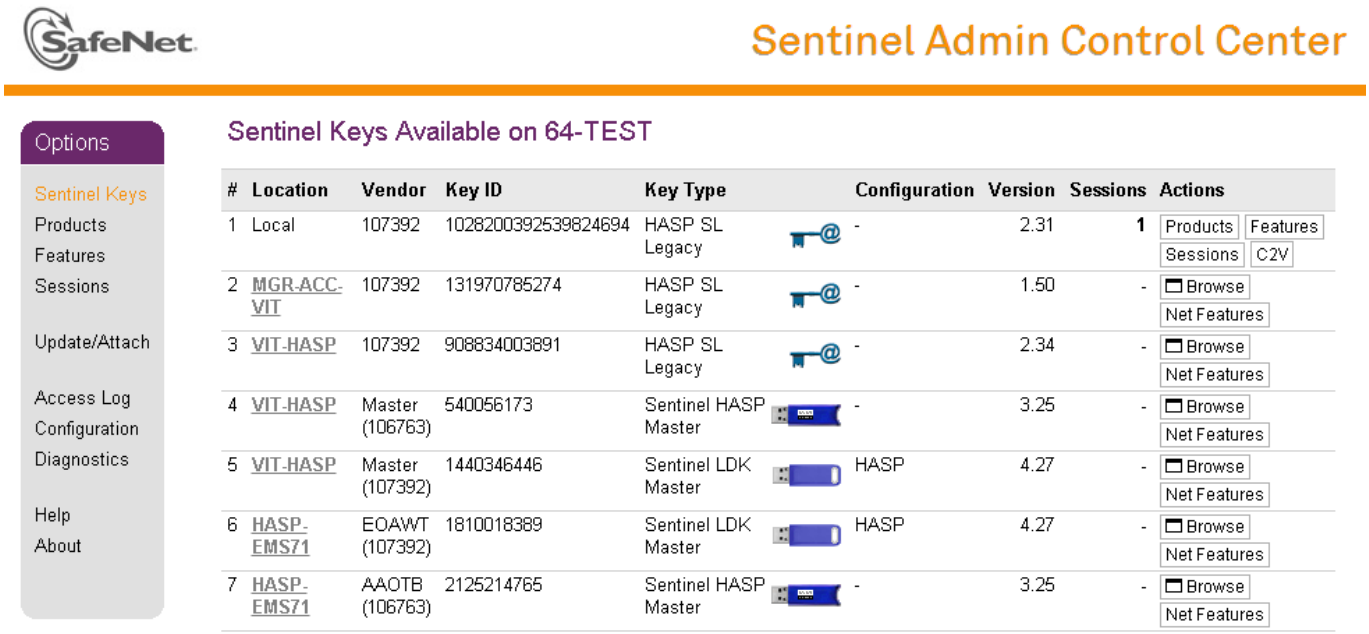
The following features of **Admin Control Center** are most often used during trial or after purchasing **AutoCode XProtect** modules:

- viewing information about all license keys currently available on local servers;
- viewing information about all license key components as well as limitations associated with them.

Sentinel Keys tab

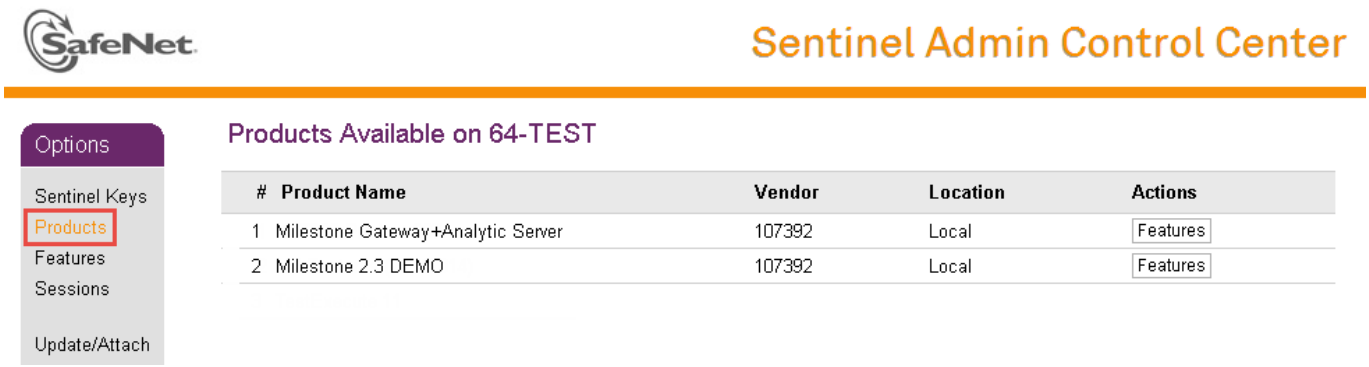
This page shows the license keys that are currently in the system including local keys — they have the **Location** parameter set to **Local**. The most common features are as follows:

- The **Sessions** button opens a page that contains the information about the sessions of a selected license key. A session is a request that the licensed application makes to a license component. If the session is not displayed, the respective functionality is not available.
- The **Blink on/off** button enables/disables blinking for an LED of a specific hardware key allowing for faster identification of the key.
- The **Features** button shows all features of a local license key.



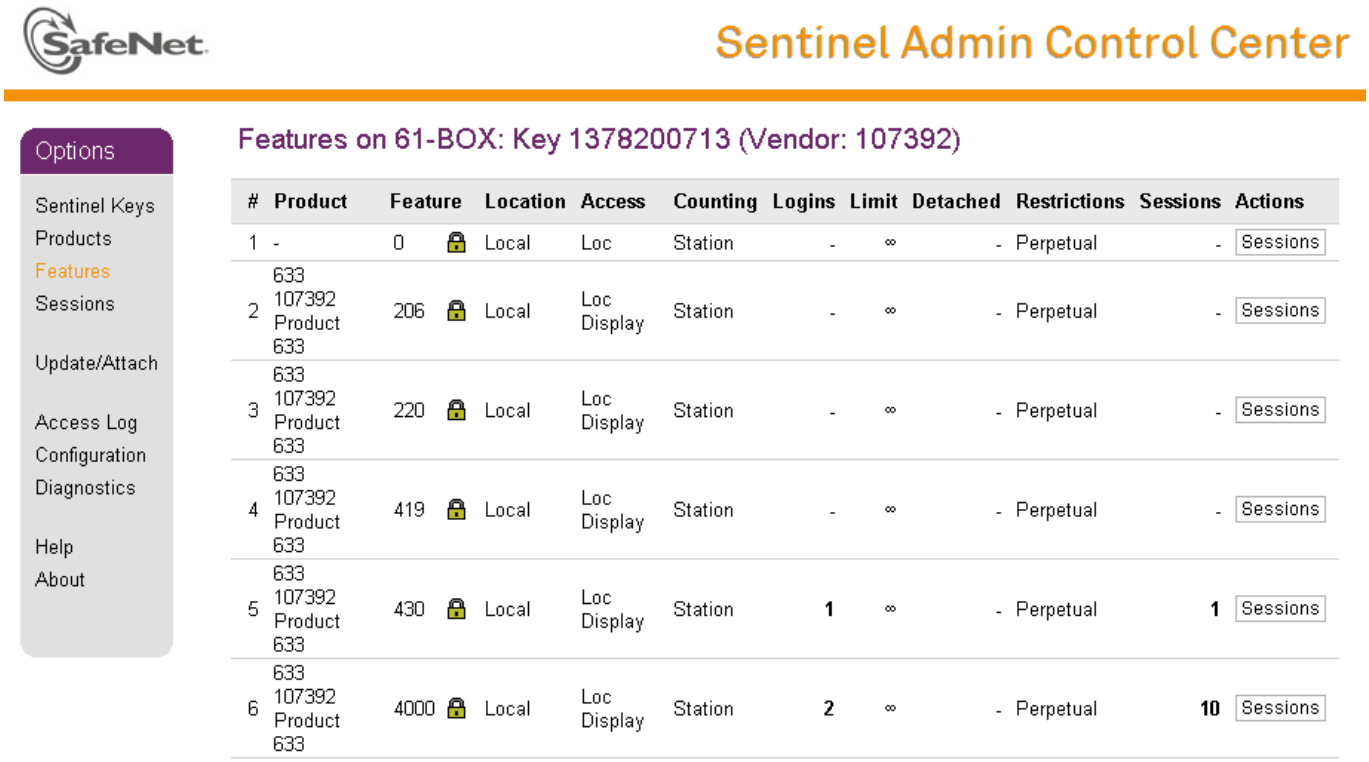
Products tab

This page displays the names of all products (**Product Name** column) available under all license keys in the network. However, **AutoCode XProtect** modules can only function under the license keys that have been installed locally.



Features tab

This page displays all the products included into the firmware of a specific local key. It also allows checking license restrictions and provides information about current activities on each feature.



#	Product	Feature	Location	Access	Counting	Logins	Limit	Detached	Restrictions	Sessions	Actions
1	-	0	Local	Loc	Station	-	∞	-	Perpetual	-	Sessions
2	633 107392 Product 633	206	Local	Loc Display	Station	-	∞	-	Perpetual	-	Sessions
3	633 107392 Product 633	220	Local	Loc Display	Station	-	∞	-	Perpetual	-	Sessions
4	633 107392 Product 633	419	Local	Loc Display	Station	-	∞	-	Perpetual	-	Sessions
5	633 107392 Product 633	430	Local	Loc Display	Station	**1**	∞	-	Perpetual	**1**	Sessions
6	633 107392 Product 633	4000	Local	Loc Display	Station	**2**	∞	-	Perpetual	**10**	Sessions

5 RECOGNITION SETTINGS

general information

The process of setting up LPR is comprised of two main stages:

- **Stage 1.** Configuring the recognition in **Management Client**. This operation is possible after installing the **Configurator** module.
- **Stage 2.** Requesting configuration data from the **Configurator** module and transferring them to LPR sub-system. This operation is performed using the **Analytic Server** service. The service is to operate on the same host as the LPR sub-system.

The algorithm for configuring the recognition using **Configurator** module (Stage 1) is comprised of the following actions:

1. Creating and configuring video source servers (see the Video Source Servers section).
2. Creating and configuring servers for receiving the events (see the Target Servers section).
3. Creating recognition servers (see the Recognition Servers section).
4. Creating the channels based on recognition servers that are being used (see the Channels section).
5. Configuring each channel individually:
    - General Settings Tab
    - Recognition Settings Tab
    - Settings of Events tab
    - Settings of Dynamics tab
    - Additional Settings tab

Within the UI of **Management Client**, the features of LPR sub-system are represented as system objects combined into **VIT Analytic Service** group within the **MIP Plug-ins** group.
The LPR process is activated and maintained (Stage 2) using **Analytic Server** service (**vitanalyticsrv**). The service is controlled by a manager application with the same name.
The UI of the manager application contains four tabs including **Analytic Server** and **XProtect**. The feature-set of these tabs is used during LPR setting up/activation. The detailed instructions can be found in the Configuring Analytics Server section.

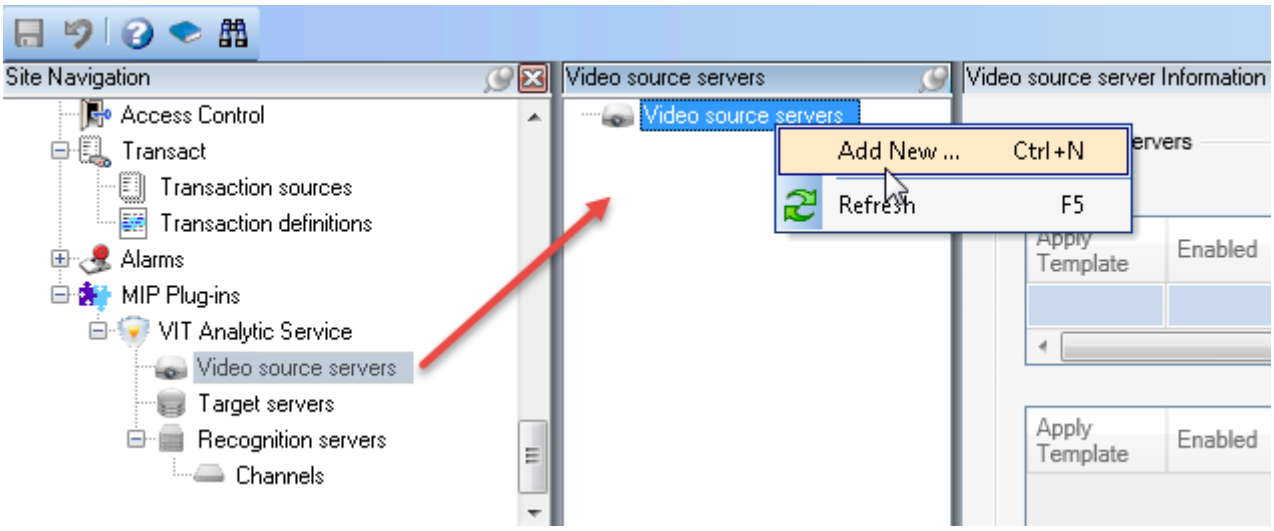
video source servers

Video source server functionality is necessary to determine which **XProtect Recording Servers** will provide data for processing.

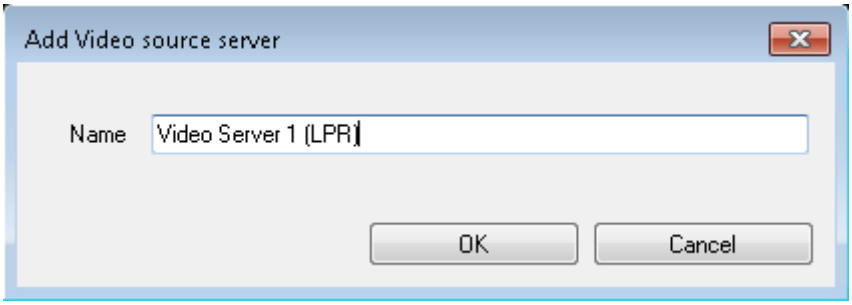
Creation

To create a video source server object:

1. Select a **Video source servers** object on navigation panel.
2. Open the context menu of the object and click **Add new** on the overview panel located to the right.



3. Provide the name for a new object in the window that appeared (e.g. 'Video Server 1 (LPR)') and then click ".



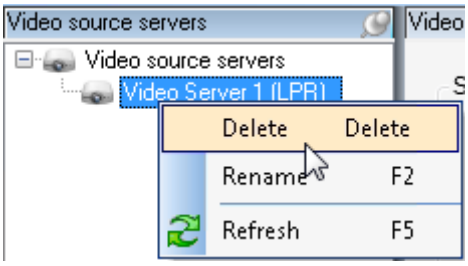
4. Save the configuration.

A new video source server object would appear on the object tree.

Deleting

To delete a video source server object:

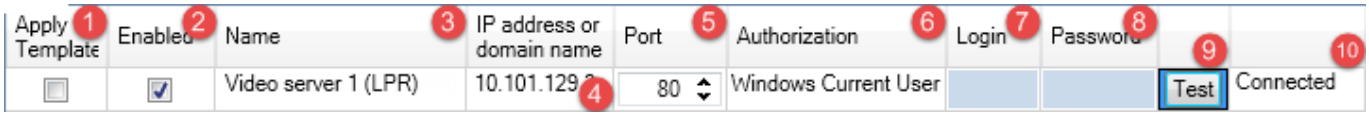
1. Open the context menu of the object to be deleted.
2. Select **Delete**.



3. Confirm deleting by clicking **OK** in the window that appeared.

Configuration

To configure a video source server object, it should be selected in the object tree and then the following parameters are to be set:



1. **Apply template**. A checkbox to apply a configuration template to this object (see the Templates section).
2. **Enabled**. A checkbox to activate the object.
3. **Name** of the object assigned to it while creating. It can be changed.
4. **IP-address or domain name** of the computer with a recording server providing the videostreams for processing.
5. **Port** of data transfer. The default value is 80: used for input/output HTTP connection between **Management Server**, cameras and client applications.

   > In **XProtect Corporate**, port for **Recording Server** object is specified when the object is created. Instructions for adding a hardware device in **Management Client** can be found in Connecting to the Videostream.

6. **Authorization** method used to connect to a server.

   > When connecting **LPR** sub-system to an **XProtect Corporate** recording server, two authorization modes are possible — **Windows Current User** (if the credentials match on both hosts) and **Windows**.

7. **Login**, to connect to a video server. If it is a domain user, the login is to be provided as 'Domain name\username' (e.g. 'HOST-64\admin'). If the user is local, the 'Username' format can be used.
8. **Password** used for connecting to a video server.
9. **Test** button to check connection status.
10. **Connection status**. 'Connected' — success, 'Failed' — failed connection.

target servers

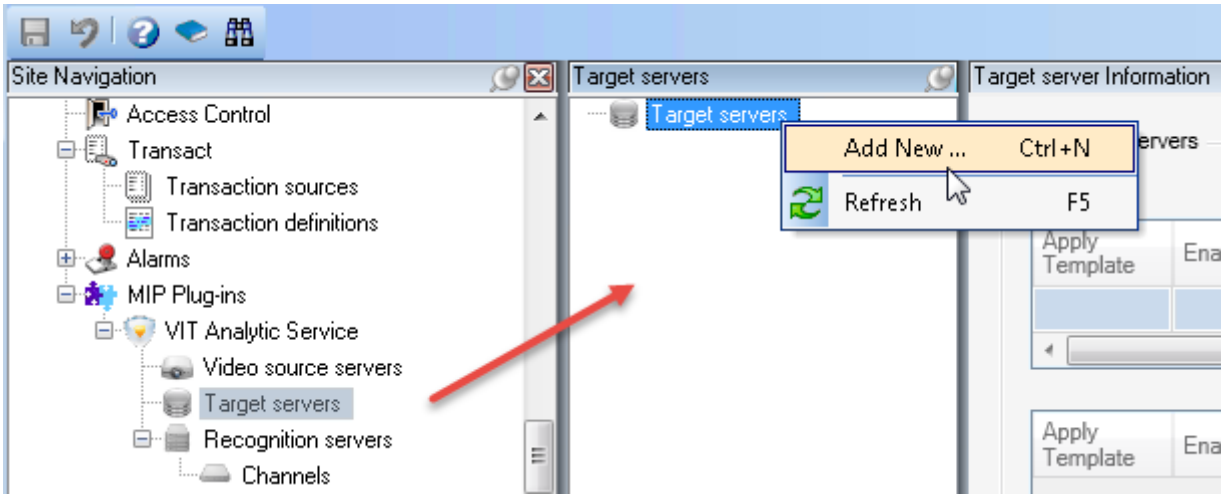
Target server functionality is necessary for selecting the **XProtect** event server to which the recognition results would be sent as analytics events.

Creating

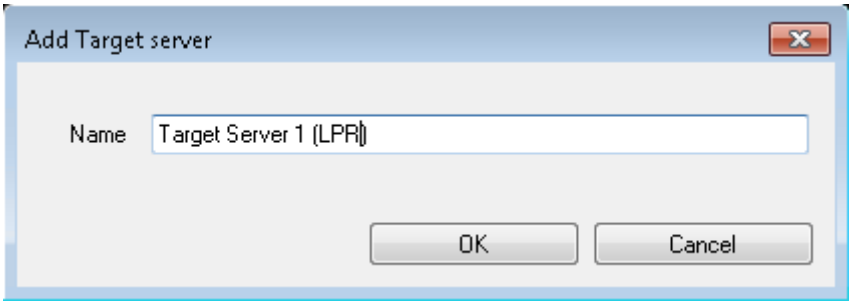
To create a target server object:

1. Select a **Target servers** object on navigation panel.

2. Open the context menu of the object and click **Add new** on the overview panel located to the right.



3. Provide the name for a new object in the window that appeared (e.g. 'Target Server 1 (LPR)') and then click ".



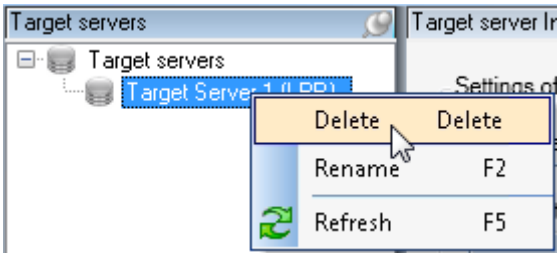
4. Save configuration.

A new target server object would appear on the object tree.

Deleting

To delete a target server object:

1. Open the context menu of the object to be deleted.
2. Select **Delete**.



3. Confirm deleting by clicking **OK** in the window that appeared.

Configuring

To configure a target server object, it should be selected in the object tree and then the following parameters are to be set:

Apply Template (1)	Enabled (2)	Name (3)	IP address or domain name (4)	Port (5)	Transport protocol (6)
☐	☑	Target server 1 (LPR)	127.0.0.1	9090	TCP

1. **Apply template**. A checkbox to apply a configuration template to this object (see the Templates section).
2. **Enabled**. A checkbox to activate the object.
3. **Name** of the object assigned to it while creating. It can be changed.
4. **IP-address or domain name** of the computer with the **XProtect** event server to connect.
5. **Port** for transferring analytics events to the XProtect Event Server (default — 9090). This value must match the value of port for receiving analytics events in **XProtect** system.

   Make sure that event transfer in **XProtect** is enabled.

   Registering Analytics Events section contains detailed instructions on registering the events and selecting port for sending them.

6. **Transport protocol** for transferring data via network (default value — TCP, alternative value — UDP).

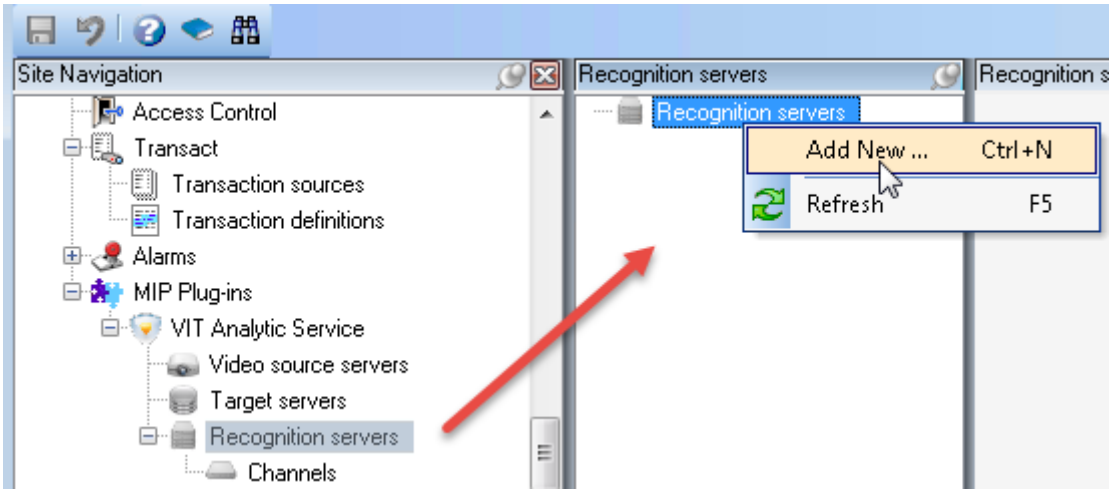
recognition servers

Recognition server object is necessary for combining recognition channels into a group to be processed on a single host.
Each recognition server object has one corresponding host used for license plate recognition (a server with **Analytic Server** and **LPR** modules installed).
One channel corresponds to one videostream. Each channel is to be configured separately.

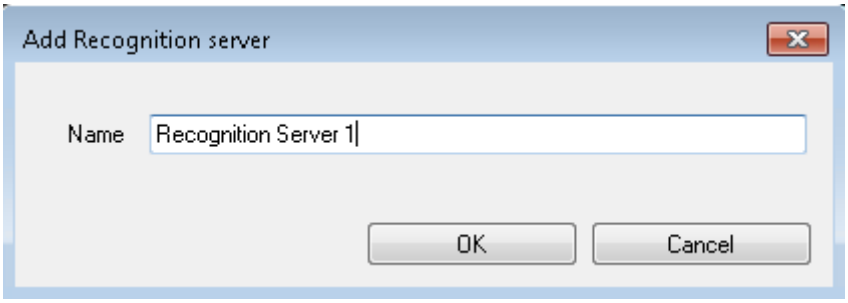
Creating and Configuring

To create a recognition server object:

1. Select a **Recognition Servers** object on navigation panel.
2. Open the context menu of the object and click **Add new** on the overview panel located to the right.



3. Provide the name for a new object in the window that appeared (e.g. 'Recognition Server 1') and then click ".



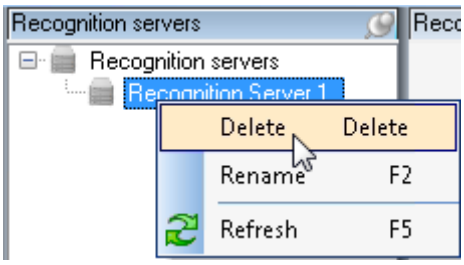
4. Save the configuration.

A new recognition server object would appear on the object tree with a **Channels** child object for further recognition configuration.
Selecting a **Recognition server** object displays its name (editable field) and a checkbox to activate the object.

Deleting

To delete a recognition server object:

1. Open the context menu of the object to be deleted.
2. Select **Delete**.



3. Confirm deleting by clicking **OK** in the window that appeared.

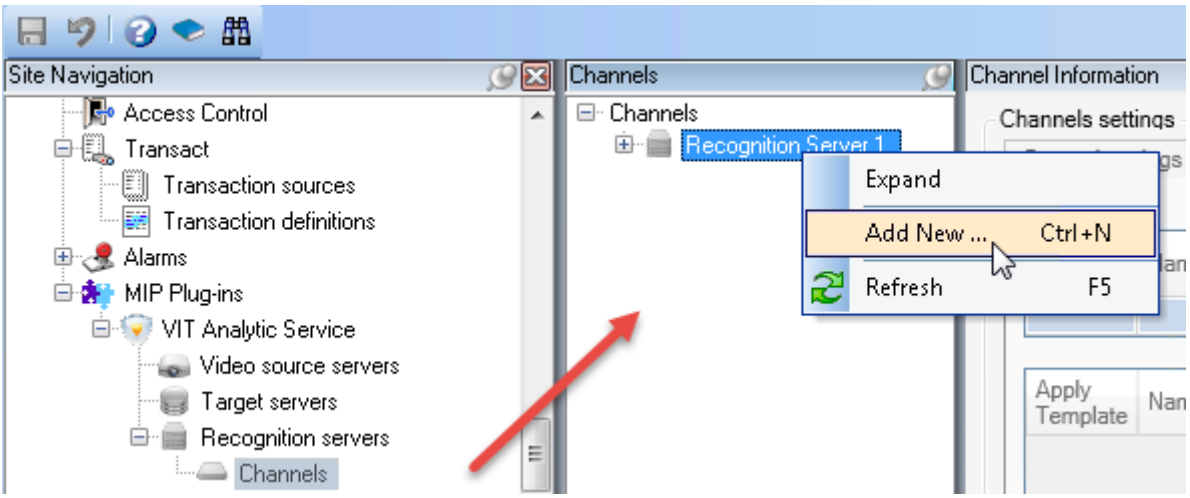
channels

In **Management Client**, a **Channel** object is used to configure license plate recognition. Therefore, one channel object represents a single videostream broadcasted live.

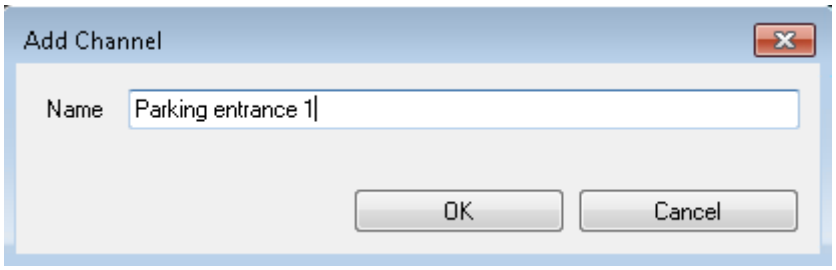
Creating

To create a channel object:

1. Select a **Recognition servers** object on navigation panel and then select **Channels** child object.
2. Open the context menu of the object and click **Add new** on the overview panel located to the right.



3. Provide the name for a new object in the window that appeared (e.g. 'Parking entrance 1') and then click ".

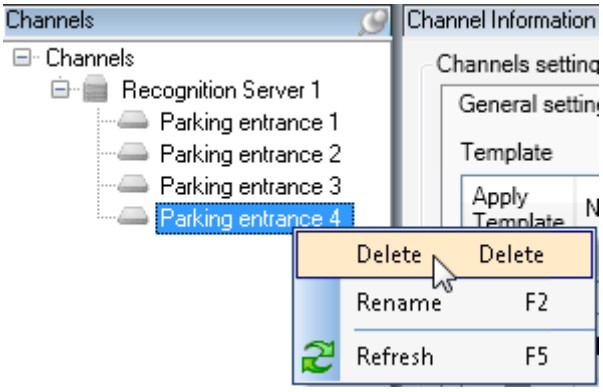


A new channel will appear on the object tree. Total number of **Channel** objects must match the number of video streams to be processed on one host (**Recognition server**).

Deleting

To delete a channel object:

1. Open the context menu of the object to be deleted.
2. Select **Delete**.



3. Confirm deleting by clicking **OK** in the window that appeared.

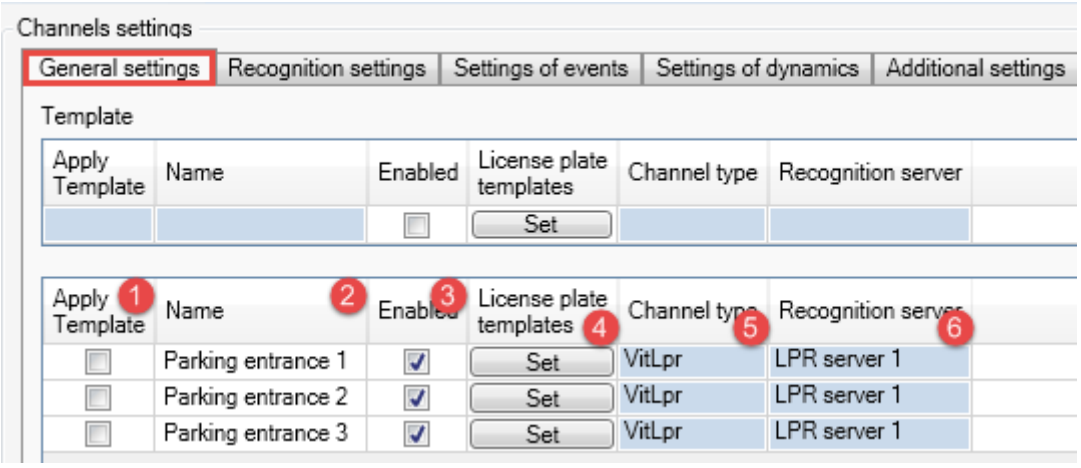
Configuring

The following sub-section contains descriptions of parameters that are set/adjusted separately for each recognition channel. In UI, the parameters are grouped based on their function. **General settings**, **Recognition settings** and **Settings of events** tabs contain mandatory parameters while **Settings of dynamics** and **Additional settings** tabs contain additional and optional parameters.
Properties panel contains settings for all channels of a single recognition server. Select a channel on the object tree to access this section and carry out the configuration.

General Settings Tab

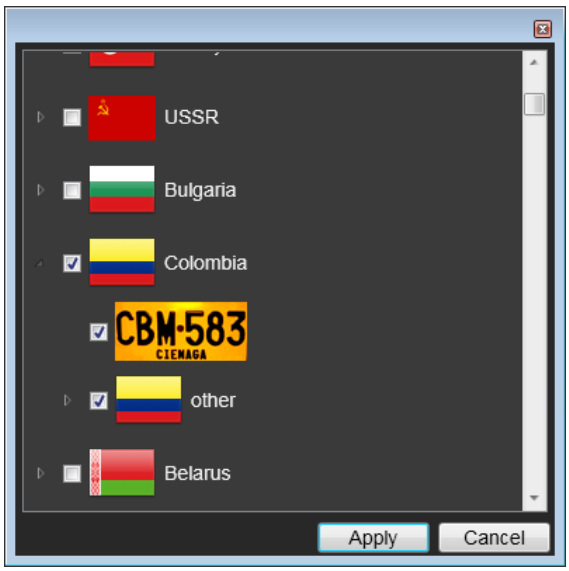
The tab provides the following features:

- enabling/disabling channels;
- selecting license plate templates for specific countries to perform recognition based on them.



1. **Apply Template**. A checkbox to apply a configuration template to a specific recognition channel (see the Templates section).
2. **Name** of the channel assigned to it while creating. It can be changed.
3. **Enabled**. A checkbox to activate the channel.
4. **License plate templates**. A button to open a template selection window. In the window, the templates would be grouped by issuer-state as nested lists. When the templates are selected, click **Apply**.

For each channel, only the templates included into the license key are available. Any number of templates can be chosen from the available set. However, it is not recommended to include the templates of license plates that would not be present on the videostream as it may affect the quality of recognition results.



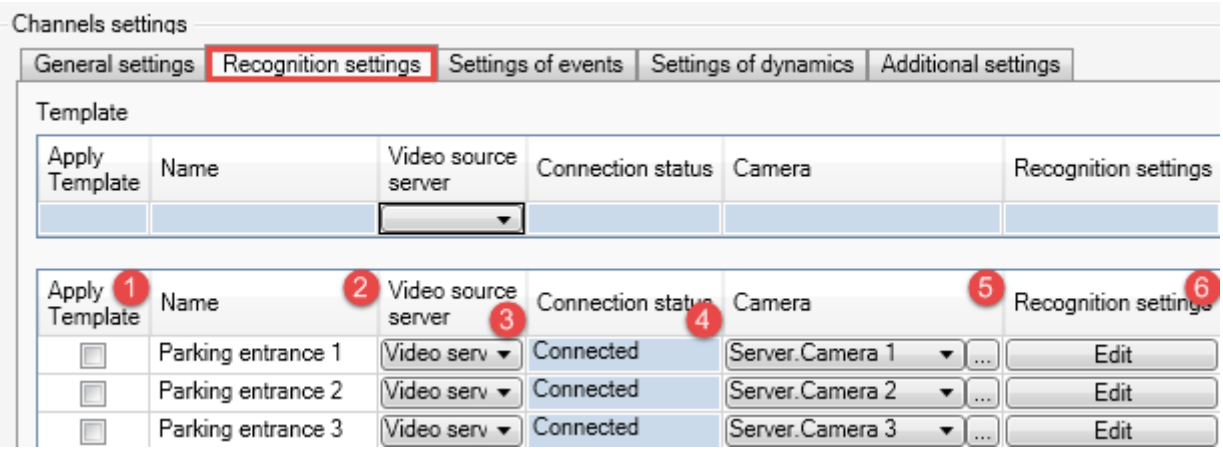
5. **Channel type**. he type of video analysis applied to the channel. This parameter has a default value set to VitLpr (Vit — name of the company that developed the module; Lpr — short for "License Plate Recognition").

6. **Recognition server**, to which the channel is connected. This value can't be changed.

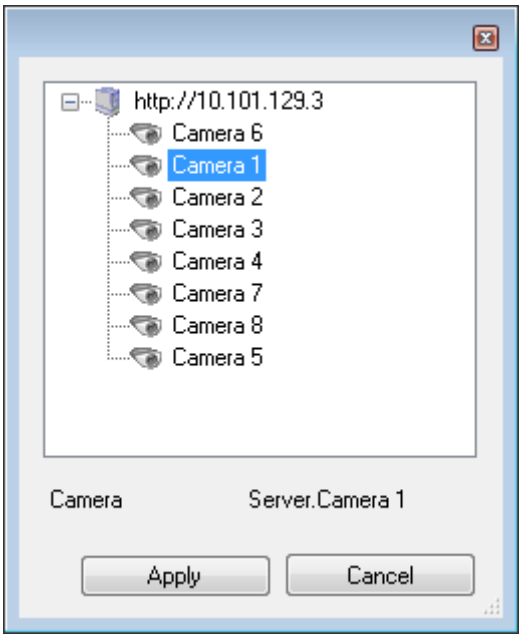
Recognition Settings Tab

The tab provides the following features:

- contains parameters for connecting to video sources (video server, camera);
- allows for setting the following key LPR parameters in a special window:
  - approximate maximum and minimum dimensions of the plate in frame,
  - target areas for detecting plates in frame (recognition zones).



1. **Apply Template**. A checkbox to apply a configuration template to a specific recognition channel (see the Templates section).
2. **Name** of the channel assigned to it while creating. It can be changed.
3. **Video source server** feeding the data for processing (see the Video Source Servers section). The source is to be selected from a dropdown.
4. **Connection status** for video server connection.
5. **Camera**. Videostream to be processed. The dropdown contains all devices connected to a selected video source server. ... button opens additional window to view the list of devices as object tree.



6. **Edit** button opens **Recognition settings** window to set key parameters of video processing.

Graphic interface of **Recognition Settings** window (Fig. 8) consists of two main sections:

- Video broadcasted by a connected source — used to set key LPR parameters with the help of graphical tools (left section).
- Panel for switching between settings and additional information display (right section).

To save the settings selected in **Recognition Settings** window, don't forget to click **Apply**.

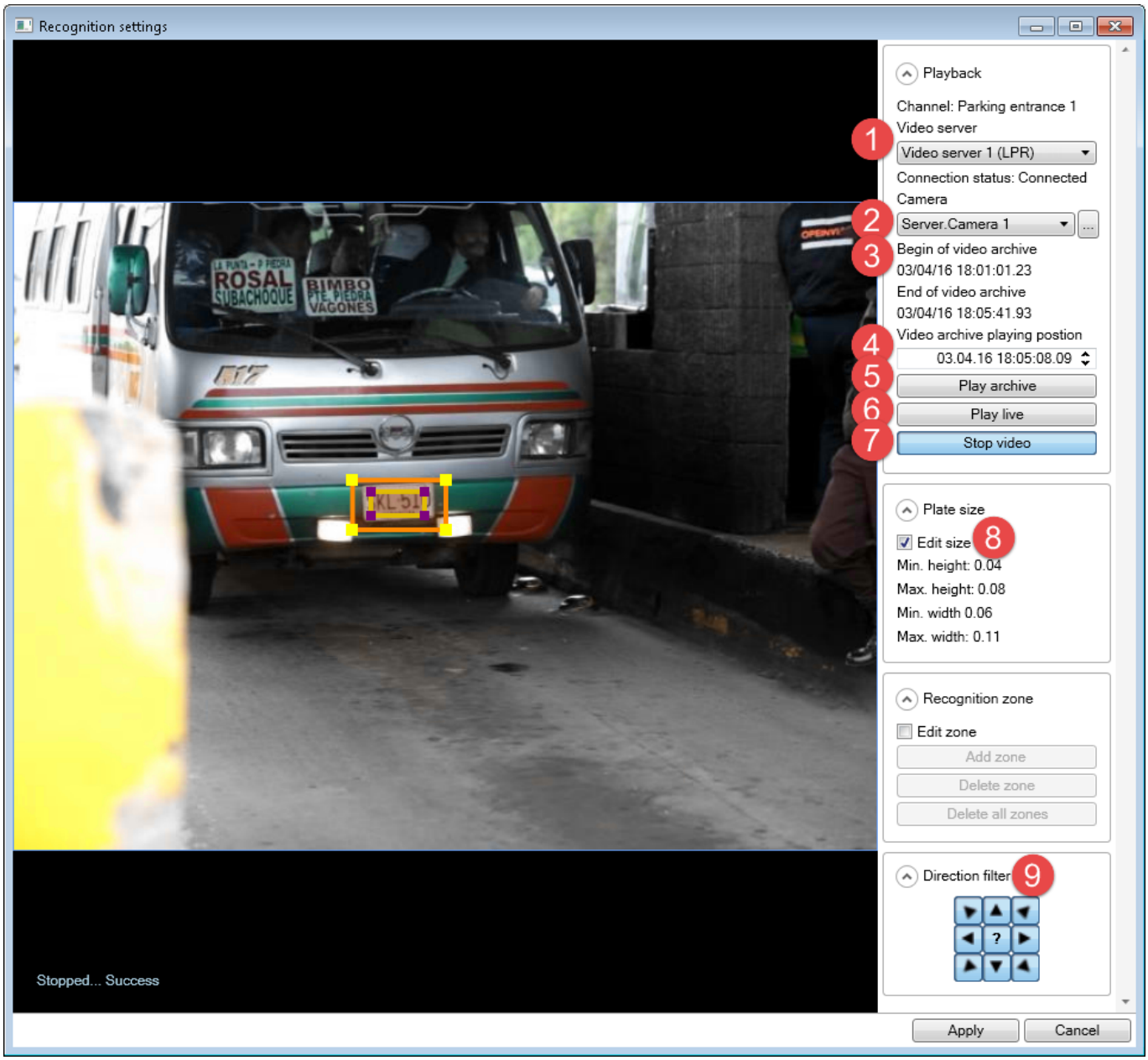


Figure 8: Graphic Interface of **Recognition Settings** window

1. **Video server** feeding the data for processing. Duplicates the **Video server** setting on channels properties panel (**Recognition Settings** tab).

2. **Camera**. The videostream from selected source. Duplicates the **Camera** setting on channels properties panel (**Recognition Settings** tab).

3. **Begin/End of video archive**. Time limits of recorded video. The values are set in following formats:

   - dd/mm/YY
   - h:min:sec.msec

4. **Video archive playing position**. The mark from which the video archive playback in the **Recognition settings** window would start.

5. **Play position**. A button to play archived video starting from the mark specified in the previous setting.

6. **Play live**. A button to play live videostream.

   > The **Play archive** button may be useful in situations when there's not much traffic in control area and, in order to speed up the configuration, a segment of video that contains license plates may be played.

7. **Stop video**. A button to stop the video when the plate is correctly positioned (fully visible) in the frame. This allows for selecting detection areas and their dimensions more precisely.

8. **Edit size**. Activating a graphical tool to set supposed maximum and minimum size of the plate in frame. After selecting the checkbox, manually set the dimensions of rectangles that appear on the videostream playback in left section of the window.

   - The frame with yellow corners represents *maximum size* of the license plate:

     > Recommended dimensions of this frame:
     >
     > – 110-200% of average license plate in the frame.
     > – 150% or less from maximum size of the license plate in the frame.

   - The frame with purple corners represents supposed minimum size of the license plate in the frame.

9. **Direction filter**, designed for cases when vehicles on the frame are moving in various directions. This may create a situation when recognition is to be performed only on vehicles moving in specific direction (e.g. on two-way access points). Only the results for such objects would be recorded into the database.

   - Each filter button represents specific direction of license plate movement in the frame.
   - Any number of directions may be selected in this setting. By default, all directions are selected.
   - In cases when direction can't be determined, the "Not determined" value is set. To have the license plates of vehicles with undetermined direction recorded into the database, the central button ? should be activated.

> **Recognition zone** (Fig. 9, settings 1, 2, 3, 4) is an area in the frame that may contain a license plate. Each channel may have up to 20 recognition zones.

This feature:

- allows speeding up the recognition by excluding the areas that are certain to not contain an image of a license plate or the areas not necessary for recognition;
- reduces the number of false events.

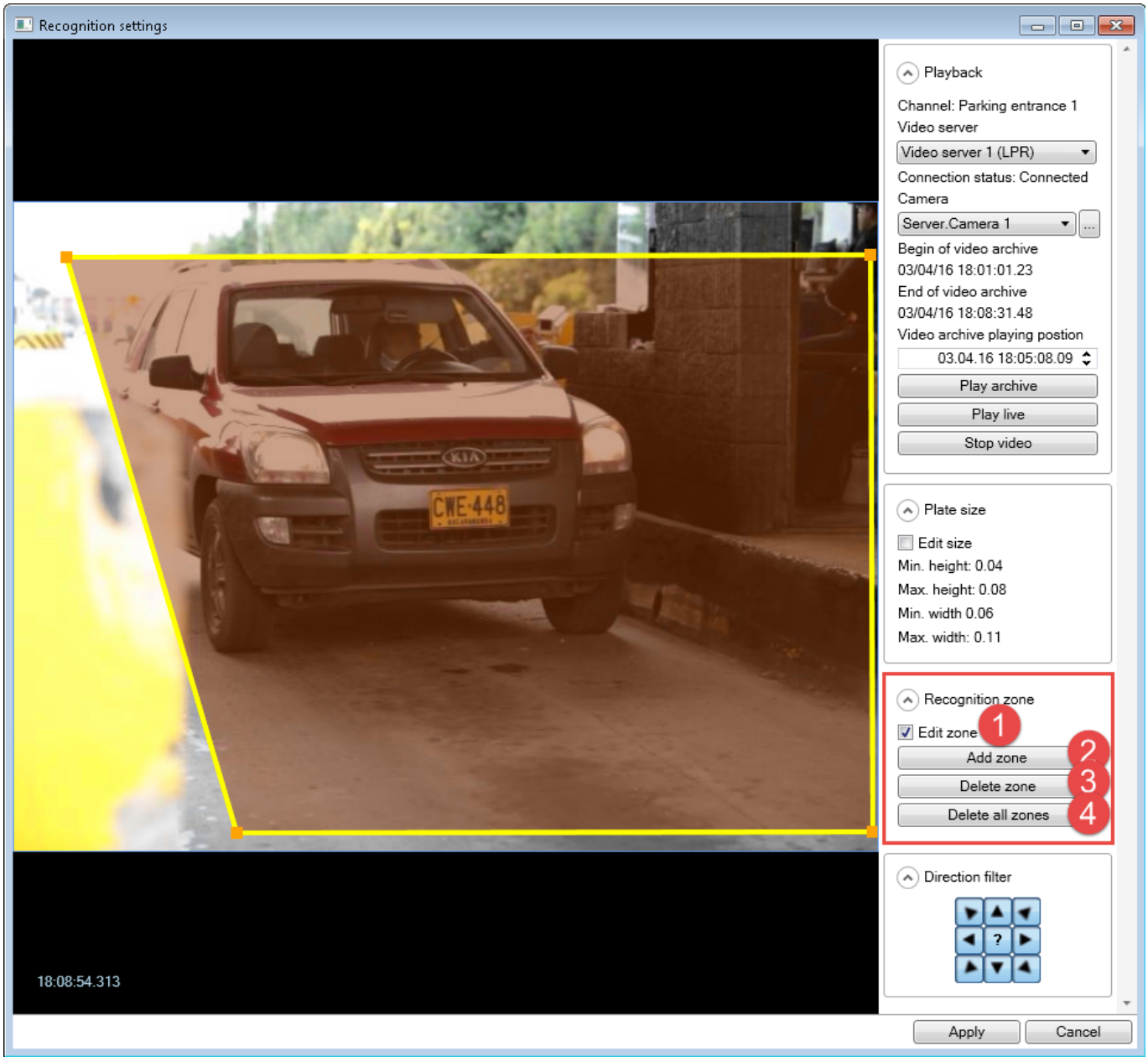


Figure 9: The process of creating a recognition zone

To *create* a recognition zone (Fig. 9):

1. Select the **Edit zone** checkbox in **Recognition settings** window (1).
2. Click the **Add zone** button (2).
3. Place the yellow handles located in corners of the recognition zone in a way to cover the area that the license plate can be found in. To add additional handles, double-click on the corresponding border of the area.
4. Click **Add zone** button again to create additional recognition zone (if needed) (Fig. 10). Adjust the newly created area as described in Step 3 above.

> it is important that there are no high-contrast objects in the frame (e.g. billboards, trees, fences). They increase the number of false events (wrong detections) during LPR. If this condition is impossible, multiple recognition zones must be created. In this case the recognition algorithm would only be applied to specific areas of the frame.

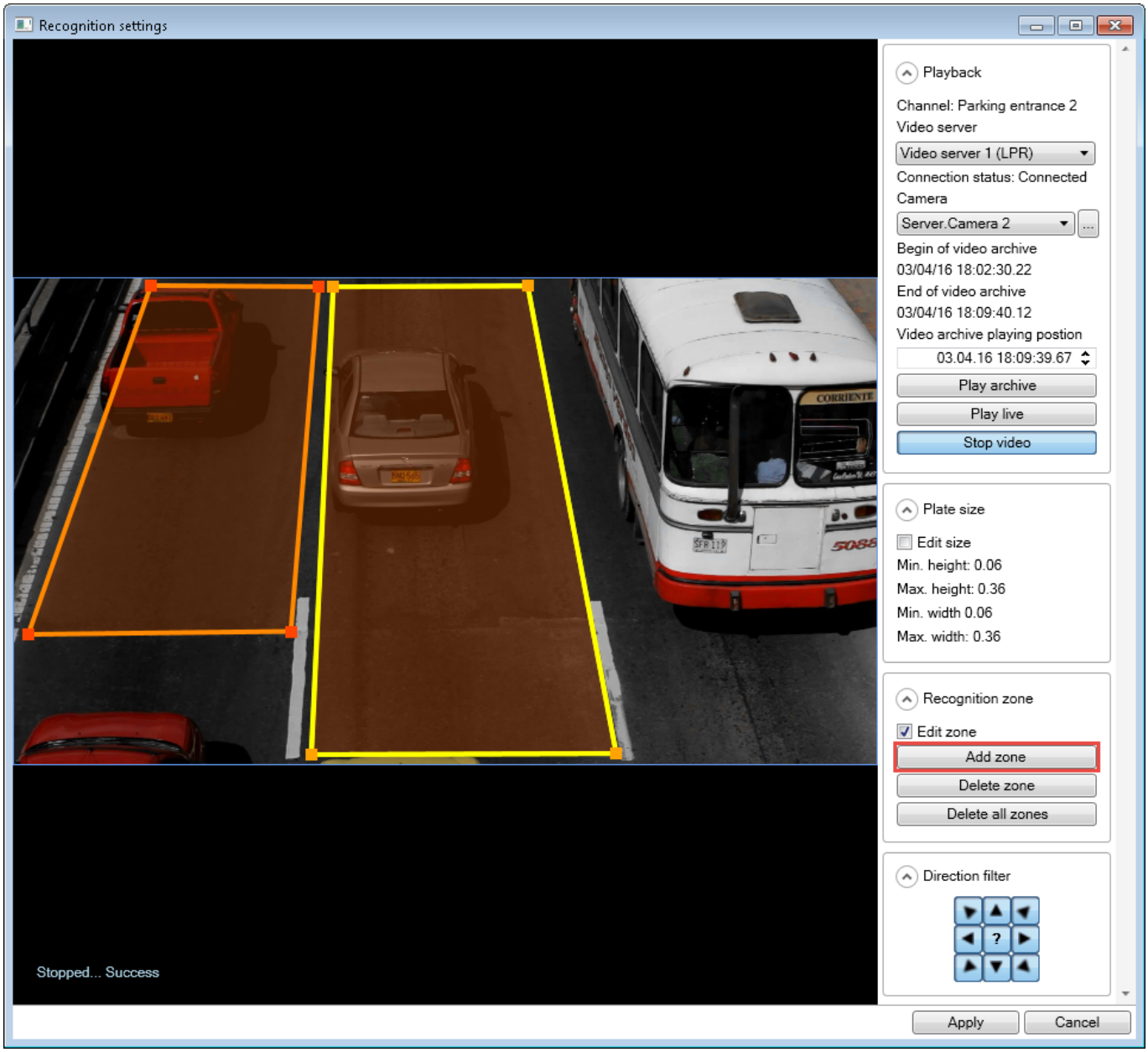


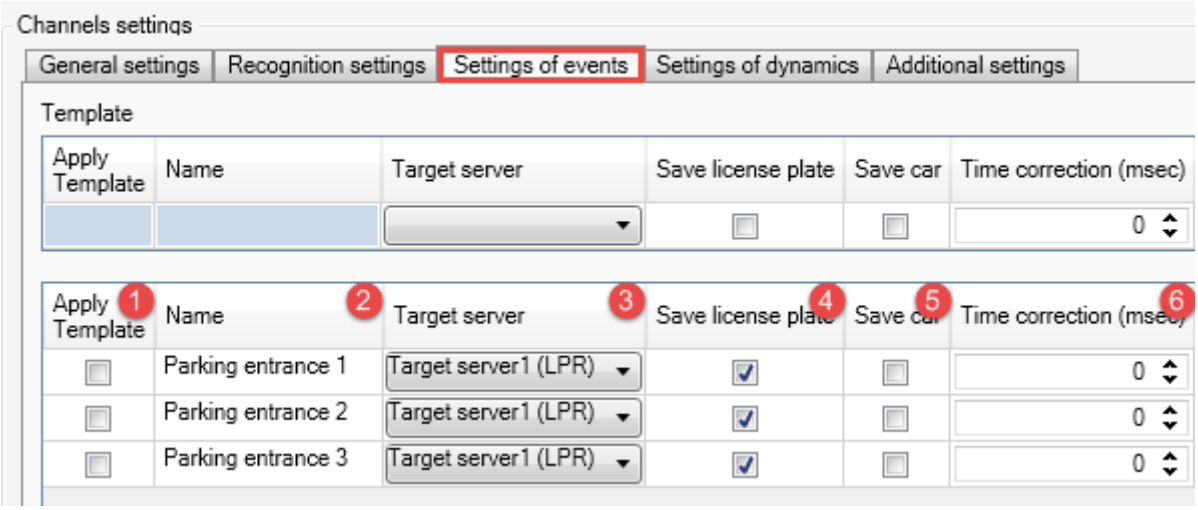
Figure 10: Creating multiple recognition zones

You can perform other operations with recognition zones:

- Right-click an existing recognition zone in order to activate and *edit* it. When the recognition zone is active, the yellow handles are available at its corners for editing the zone.
- To *delete a recognition zone*, activate it and click the **Delete zone** button (Fig. 9, 3).
- To *delete all recognition zones*, click the **Delete all zones** button (Fig. 9, 4).

Settings of Events tab

This tab enables the user to select an event server to receive the recognition events.



1. **Apply Template**. A checkbox to apply a configuration template to a specific recognition channel (see the Templates section).

2. **Name** of the channel assigned to it while creating. It can be changed.

3. **Target server**. Event server to which the recognition events will be sent (**XProtect** host with event server).

4. **Save license plate**. A checkbox to include a cropped image of the license plate into the data sent and stored in the database as a recognition event.

5. **Save car**. A checkbox to include an image of the vehicle into the data sent and stored in the database as a recognition event.

6. **Time correction (msec)** is set in cases when, upon viewing the event in client application, the position of the license plate frame does not correspond to the position of the license plate on the video (Fig. 11).

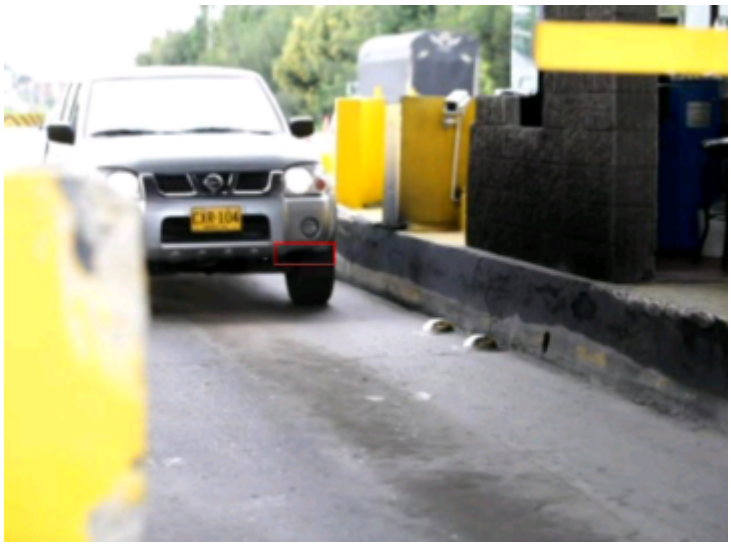
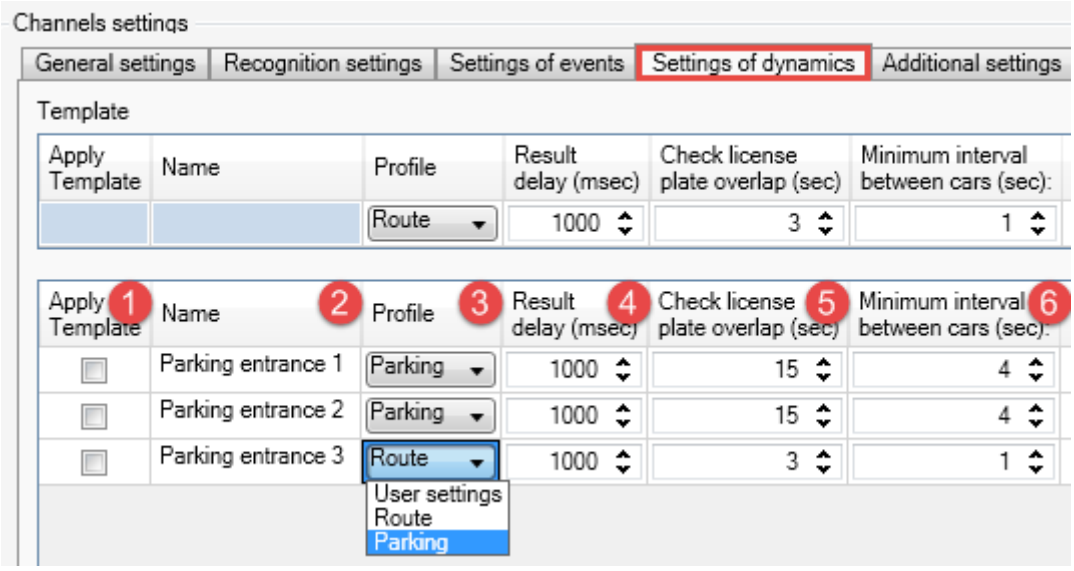


Figure 11: An example of an image where the license plate frame does not cover the license plate

> Activating **Save license plate** and **Save car** settings may cause a rapid decrease of available space in the database. When these settings are used, a specific configuration of data recording is necessary.

Settings of Dynamics tab

This tab contains dynamics settings — the settings for processing the license plates on a set of frames and providing the best outcome.



1. **Apply Template**. A checkbox to apply a configuration template to a specific recognition channel (see the Templates section).

2. **Name** of the channel assigned to it while creating. It can be changed.

3. **Profile** that, when selected, assigns optimal pre-set values to the parameters on the tab. The following profiles are available for selection:

   - **Parking** for events with vehicle speed *under* 20 kmph.
   - **Route** for events with vehicle speed *over* 20 kmph.
   - **User settings** for manual selection of optimal configuration.

4. **Result delay (msec)**. Time difference between the moment the vehicle appeared in the frame (first recognition result — **Car detected** event) and the moment best recognition result is received (**Plate detected** event). If the **Send event while car in frame** setting is enabled, the result is received after the time period specified in this setting.

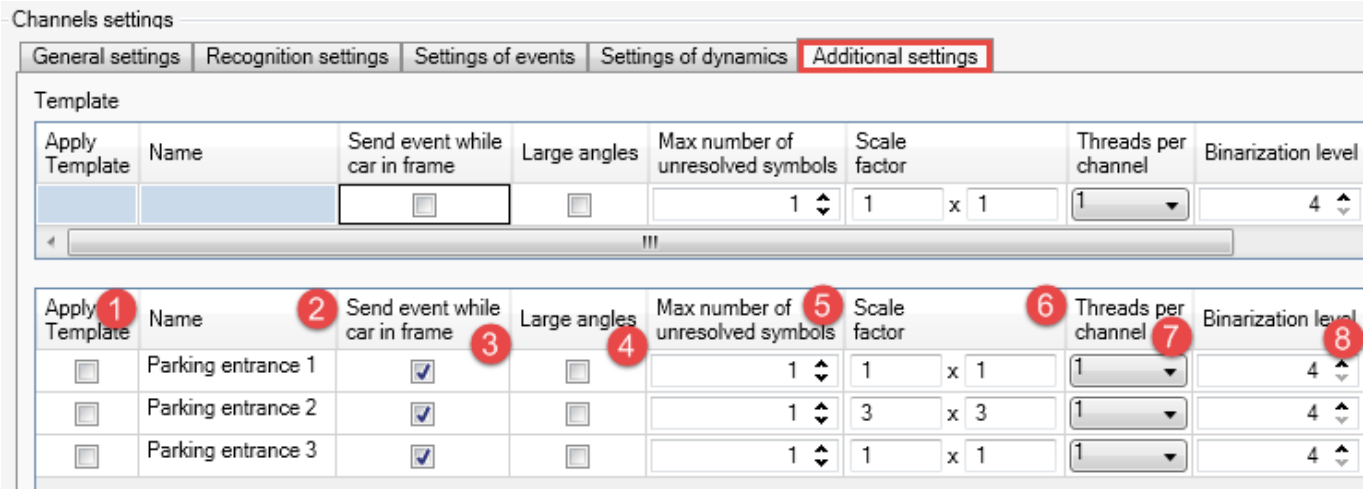
5. **Check license plate overlap (sec)**. Maximum time of the license plate being covered by another object (or beyond the camera range). If the license plate does not reappear in the frame again within the selected period, the object will be considered lost and the recognition of this object will end (**Car lost** event).

6. **Minimum interval between cars (sec)**. The time period to determine if two recognized license plates belong to the same vehicle. If identical recognition results have been received outside of the specified time period, the license plates are considered to belong to different vehicles. This setting is used to filter out duplicates if the vehicle takes a long time in the frame.

Additional Settings tab

The tab provides the following features:

- allows setting timing and quality of recognition results;
- allows adjusting parameters of recognition algorithm.

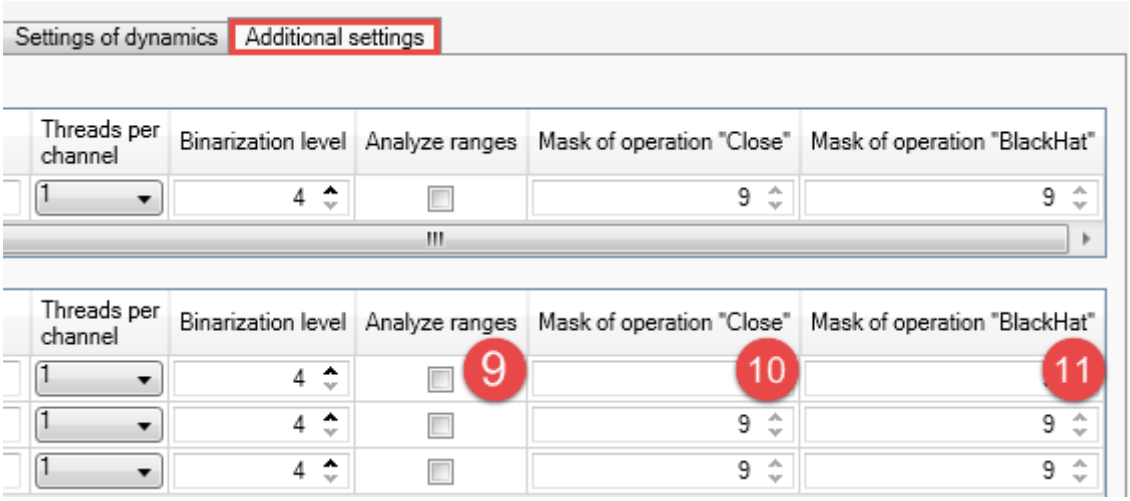


1. **Apply Template**. A checkbox to apply a configuration template to a specific recognition channel (see the Templates section).
2. **Name** of the channel assigned to it while creating. It can be changed.
3. **Send event while car in frame**. This setting is disabled by default and the recognition result is sent after the vehicle has left the frame. If this feature is enabled, the results will be sent after the time period specified in **Result delay (msec)** setting (**Settings of dynamics** tab).

   > **Send event while car in frame** setting is to be used in cases when the vehicle is in the frame for long periods of time (e.g. at access points).

4. **Large angles** checkbox is to be checked if the camera is located in a way that the license plates are placed at a 5 or more degrees angle in the frame. This feature compensates the distortion of symbols on the image and provides better recognition quality.
5. **Max number of unresolved symbols** The maximum amount of unrecognized symbols acceptable to allow for a recognition event to be generated. Unrecognized symbols will be replaced by an asterisk (*) symbol. If the license plate contains more unrecognized symbols that are specified in this setting, it will not be sent as a recognition event.
6. **Scale factor**. Scaling parameters specify the extent to which the source image will be condensed in width and height before license plate detection. This feature allows enhancing performance of the recognition module. Video is converted according to the following formula: W/x, H/y (W and H stand for width and height; x and y stand for values provided by the user). This setting allows for:
   - Increasing the performance of the sub-system by decreasing the detection time (the setting should be within 1-10 range).
   - Compensating for proportions warping in provided frames (the setting should be within 0.01-0.99 range).

> Changes in scaling are not reflected in the UI.



7. **Threads per channel**. Breaking recognition process into multiple threads (parallel processes) and then combining their results. The number of such processes is specified in **Threads per channel** setting. it is recommended that the number of threads in this setting equals the number of lanes in the frame. For example, if the camera oversees three traffic lanes, the recognition process for corresponding channel should be divided into three threads.

> If CCTV uses analog cameras or cameras with resolution lower than 1.3 MPix, the acceptable value for a **Threads per channel** setting is 1. However, this value is not suitable in cases when the recognition takes place on highways and uses cameras with resolution larger than 1.3 MPix. In such cases, processor gets excessive workload and can't process required amount of frames. Cases like this require splitting recognition process into multiple processes using **Threads per channel** setting.

> If the total number of threads for channels in the object list exceeds the number specified in license agreement, only the number of threads included in the license would be used. In this case, active threads would be selected randomly. In cases when additional recognition channels must be used, the user should contact the manager and request the license to be updated to include more recognition channels.

8. **Binarization level**. The value inserted into the Binarization level field represents the number of iterations in license plate detection algorithm. Higher level of binarization leads to better recognition quality, but affects the overall performance of the system. it is recommended to set a higher binarization level for recognizing slow-paced objects (for example, on parking lots). And, respectively, the lowest possible level should be set when working with license plates on highways. The default value is 4. Max value is 10.

9. **Analyze ranges**. A checkbox to activate additional operations of license plate detection algorithm. Enabling this feature increases the quality of recognition, but also significantly increases the time of this operation.

10. **Mask of operation 'Close'**. 11. **Mask of operation 'BlackHat'**. The values used by LPR algorithm when performing Close and BlackHat operations on images of license plates. This setting is to be edited by support engineers and recognition engine developers in case of poor recognition quality.

templates

To decrease the time it takes to configure LPR sub-system, the templates feature was created. A template is a preset or a combination of presets that can be assigned to object parameters. Templates are used for configuring objects that must have similar values in either all parameters or some of them allowing to avoid inserting them manually.
Templates are available for the following system objects:

- **video source servers**;
- **target servers**;
- **channels**.

The fields in template section (Fif. 12, 1) correspond to the fields of the object being configured (Fig. 12, 2).

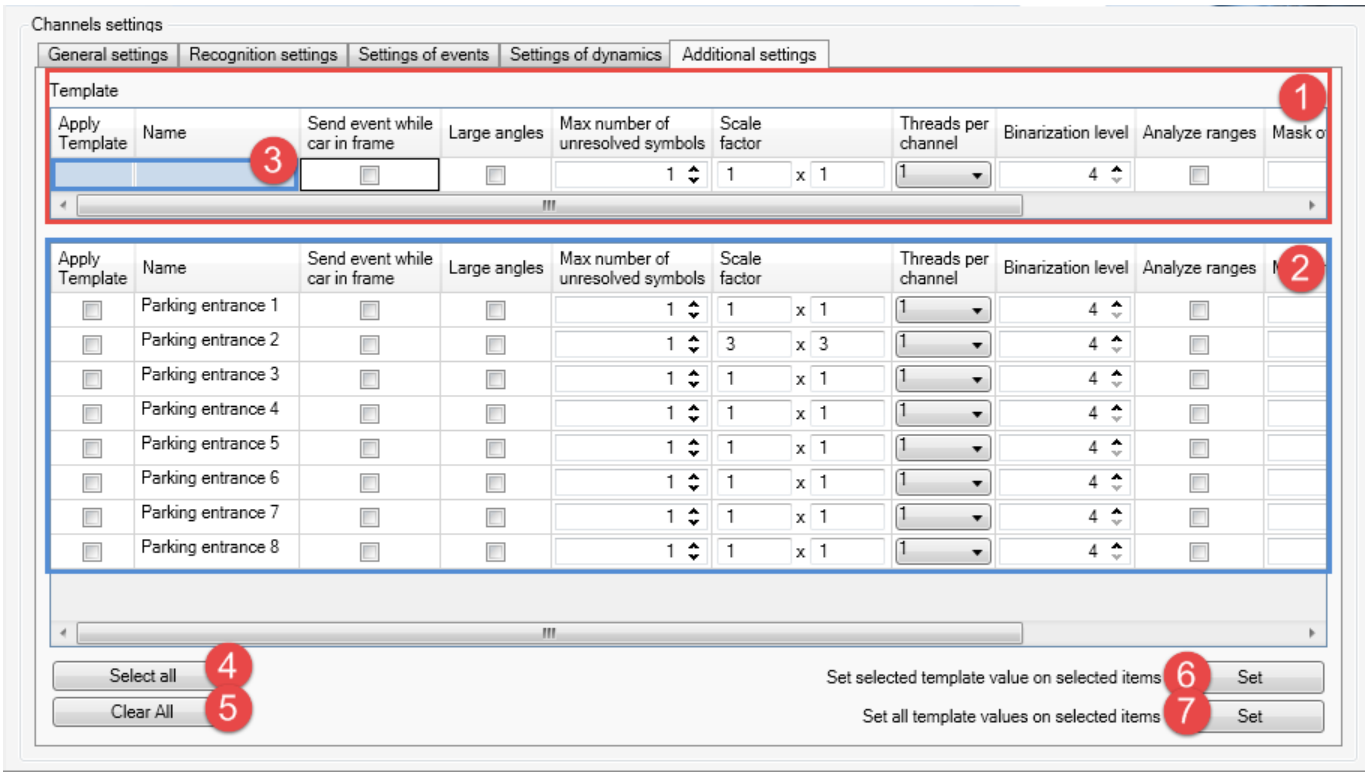


Figure 12: Template feature on the properties panel of a system object

Applying a template to a group of objects requires the following steps:

1. Determining the values that would be most often set when an object is configured.
2. Inserting the values into respective template fields (Fig. 12, 1).

   > Some object fields, such as **Name**, are supposed to be filled individually for each object and are not included into template fields. Such fields are highlighted blue (Fig. 12, 3).

3. Checking **Apply template** checkboxes for the objects that should be pre-set using a template.

   > - Click **Select all** button (Fig. 12, 4) to select all objects to be pre-set using a template.
   > - Click **Clear all** button (Fig. 12, 5) to deselect **Apply template** checkbox on all objects.

4. Applying the template to object settings:

- To apply *a single specific template value to the settings* of selected objects, highlight the template value and click upper **Set** button (Fig. 12, 6). Also, see Fig. 13.
- To apply *all template values to the settings* of selected objects, click lower **Set** button (Fig. 12, 7).

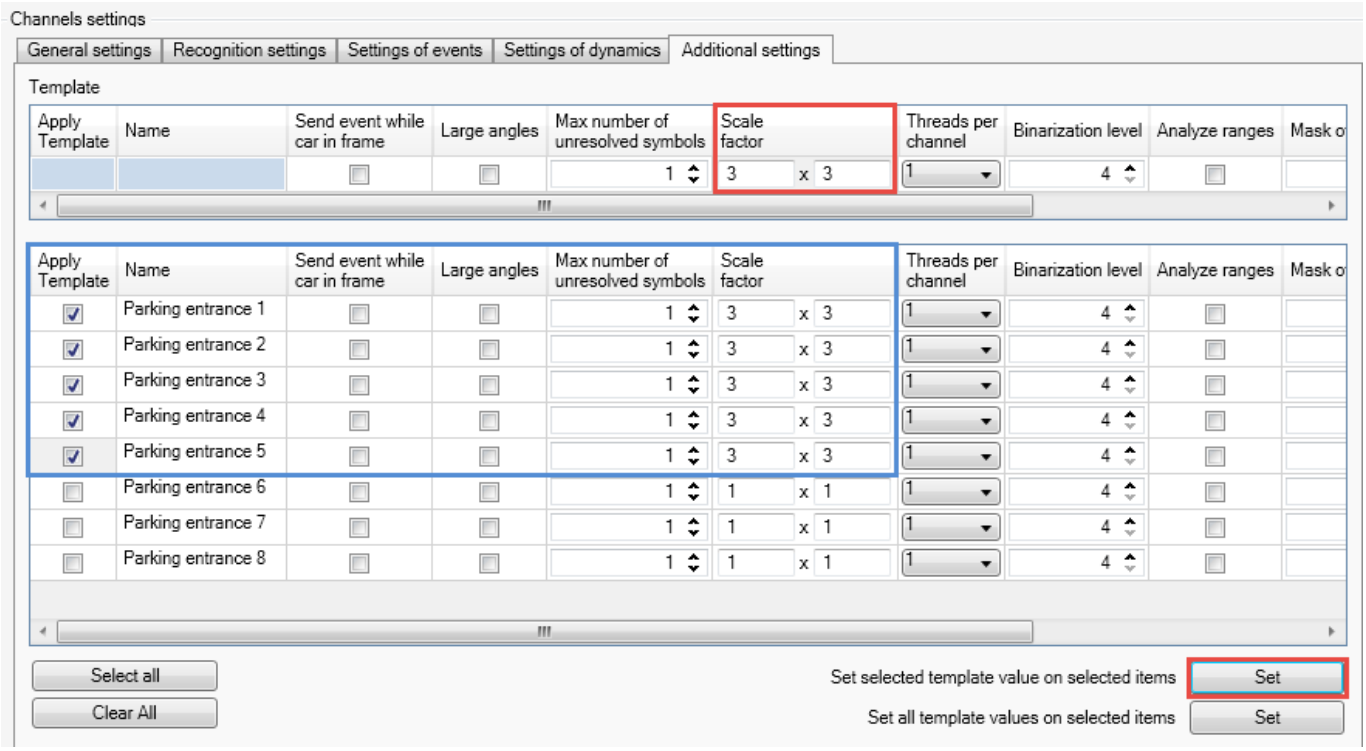


Figure 13: Order of actions to apply a created template

Fig. 14 illustrates having a template applied to **Scale factor**, **Threads per channel** and **Binarization level** settings of a **Channel** object.

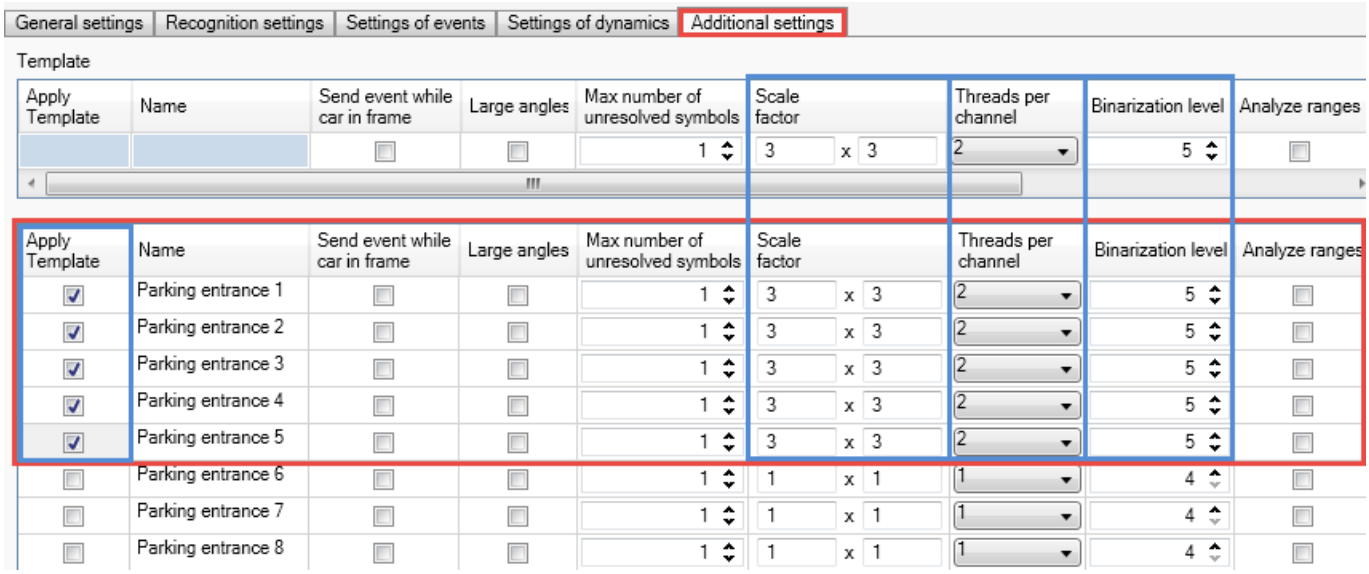


Figure 14: Example of using a template feature

configuring analytics server

License plate recognition is carried out by **LPR** module. **Analytic Server** service is necessary for activating and controlling this process as well as monitoring the state of each channel and creating a state log.
Analytic Server service is configured in two steps:

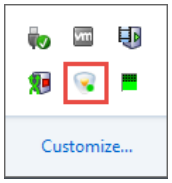
- Connecting to **XProtect**.
- Setting logging parameters.

Administration of **Analytic Server** service includes monitoring the state of recognition channels.

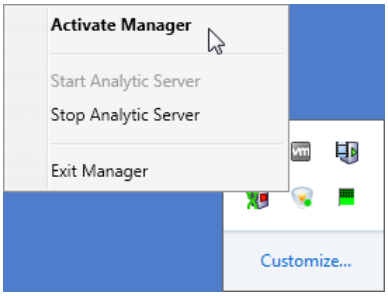
Launch and Shutdown

VIT Analytic Server (**vitanalyticsrv**) is a Windows OS service launched using standard methods of the operating system. After the management service is installed, **Analytic Server** may be launched:

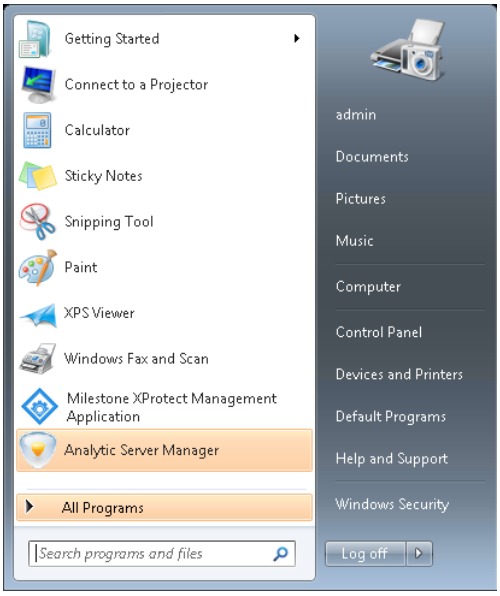
- by double-clicking the icon on the system tray.



- from the context menu of the icon.



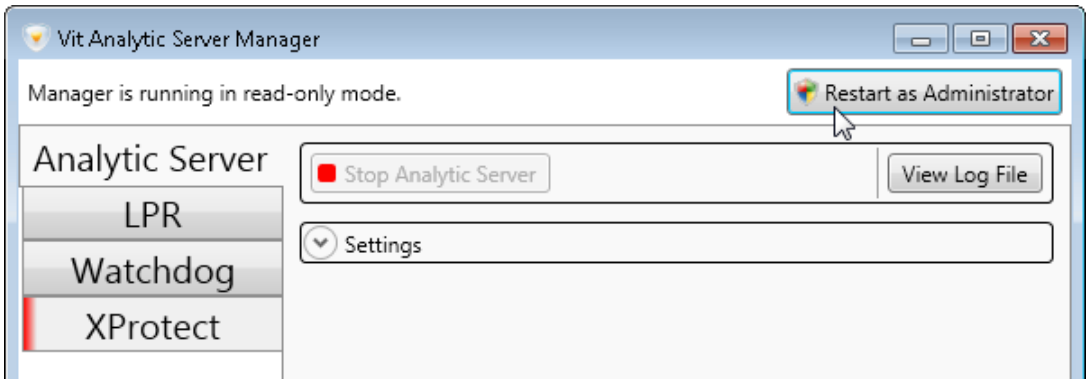
- from **Start** menu.



Depending on the state of **Analytic Server**, the icon looks different:

- — service launched.
- — service stopped.
- — service is not configured (see Connecting to XProtect).

> Configuring the **VIT Analytic Server** service is only possible if the management application of the service was launched by a user with admin rights. If this condition is not met, management application can only be used for monitoring operation of the channels (read-only mode).
> After the management application is relaunched by an admin user, it can be stopped and configured.



Connecting to XProtect

Sending configuration data to **LPR** sub-system is a necessary operation to be performed after configuring license plate recognition in **Management Client**. This requires connecting **LPR** module to an existing **XProtect** server. In order to do this (Fig. 15):

1. Open **Analytic Server** manager.
2. Go to **XProtect** tab.
3. Specify the connection settings:
   - **Server Address** of a computer with **Management Server** installed, that was used to configure the recognition on previous stages. The format to be used is 'http://server_address:port' or 'http://domain_name:por
     If 80 port is used, it is not necessary to include it.
   - **Authorization Type** used for connection: **Basic**, **Windows** or **Windows 'urrent User**.
   - **User** — a login to connect to **XProtect** server. If a domain user is used, the login should be inserted in 'Domain name\Username' format (e.g. "HOST-64\admin"). For local users, this parameter may be filled in 'Username' format.
   - **Password** used to connect to **XProtect** server.
4. Select the recognition server from which the configuration data should be sent to LPR sub-system. Click **Load Configurations** button and select the server from the dropdown.
5. Click **Save and Apply** button to connect (or click **Cancel** to discard the changes).

If the connection is successful, 'Connected' status will be shown at the top of **XProtect** tab.

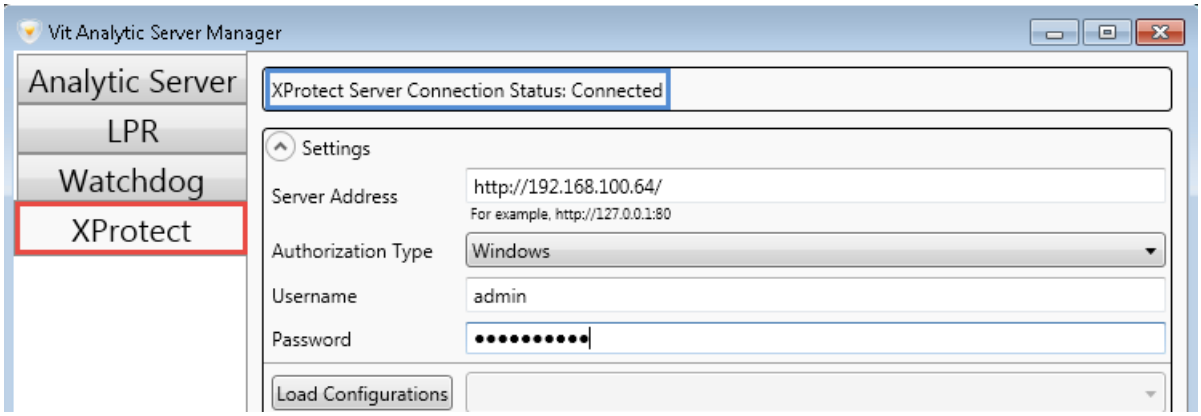
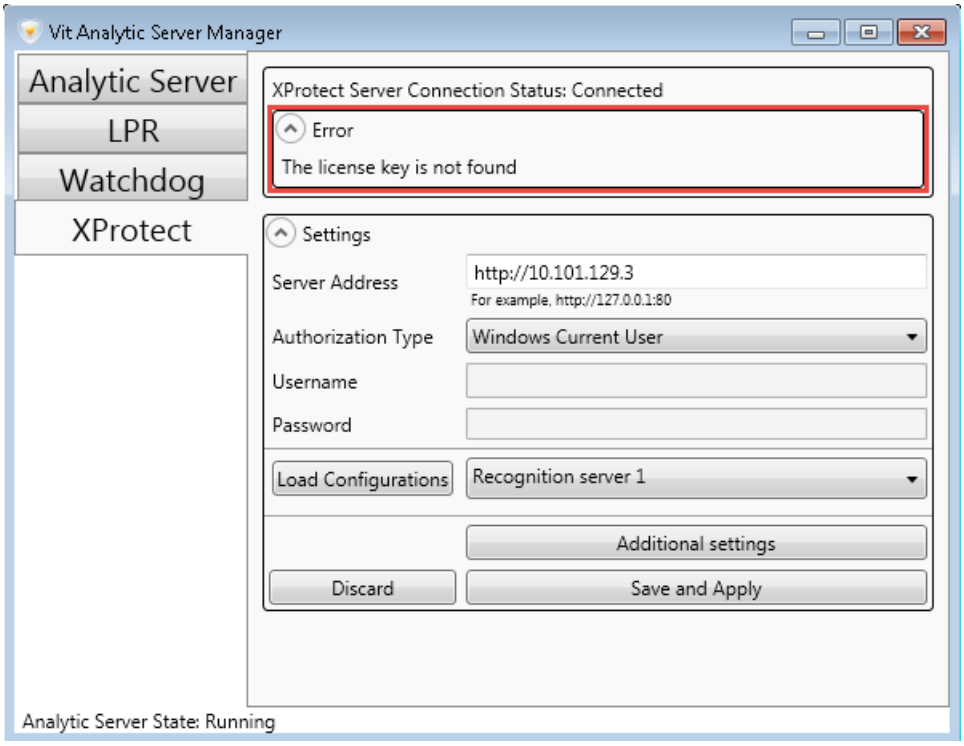


Figure 15: Connecting recognition sub-system to a configuration server

LPR tab would also display information about active recognition channels (see the Monitoring Channels section).

> **XProtect** tab may also display an error message in case of licensing error (i.e. the license key not found).



Receiving Events from Third-Party Sources

Additional settings window is opened by clicking a respective button on **XProtect** tab of **Analytic Server** manager. This window is created to configure receiving of generic events i.e. messages sent from sources outside **XProtect** system (e.g. motion sensors).

> A third-party service notifies of such an event by sending a message line to generic events port. The line is verified to meet a specific condition (described further) to enable further reactions to it.

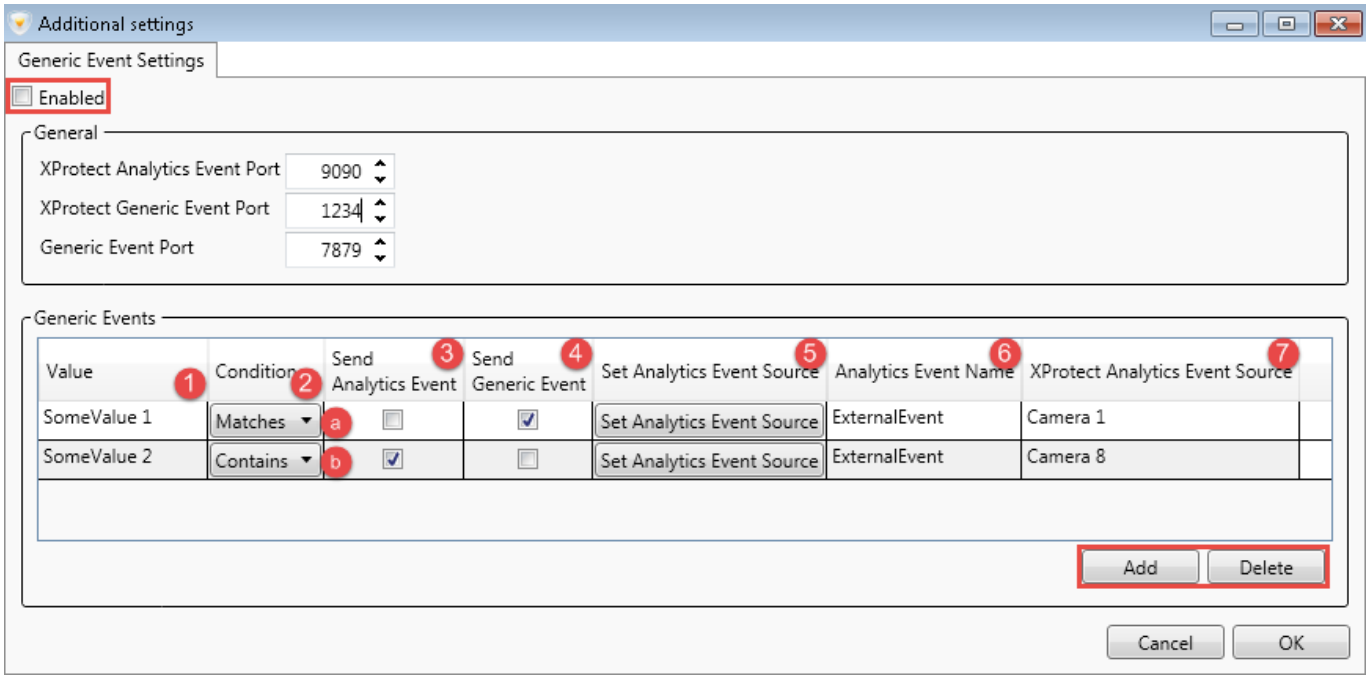


Figure 16: Receiving generic events within **Analytic Server** module

Select **Enabled** checkbox to activate this feature (Fig. 16).
The port to receive generic events is specified in **Generic Event Port** setting. Such events as well as analytic events created based on them can be redirected to **XProtect** system — in order to set this up, specify respective numbers in **Generic Event Port** and **XProtect Analytic Event Port** settings.
Default values:

- **XProtect Analytic Event Port** — 9090.
- **XProtect Generic Event Port** — 1234.
- **Generic Event Port** — 7879.

Generic events section allows specifying the generic events to be redirected to **XProtect** system:

1. **Value** — the string for checking if the events meet the condition.
2. **Condition** applied to validate incoming message strings:
   - **Contains** — the events are redirected if the string contains the value (sub-string) specified in **Value** field.
   - **Matches** — the events are redirected is the string completely matches the value specified in **Value** field.
3. **Send Analytics Event** — checkbox enables events to be redirected to **XProtect** system if they meet the condition. The event to be redirected is specified in **Analytics Event Name** field (Fig. 16, 6). After the **Set Analytics Event Source** button (Fig.16, 5) is used to select it, the name of the device sending the event is to be specified in **XProtect Analytics Event Source** field (Fig. 16, 7).
4. **Send Generic Event** checkbox enables generic events to be sent to **XProtect** system if they meet the condition.

Add button creates a set of fields necessary to configure redirecting for one event. **Delete** button removes selected line from **Generic Events** section.

Monitoring Channels

The **LPR** tab of **Analytic Server** manager application (Fig. 17) contains information about recognition channels based on their connection to video source (**XProtect** system).

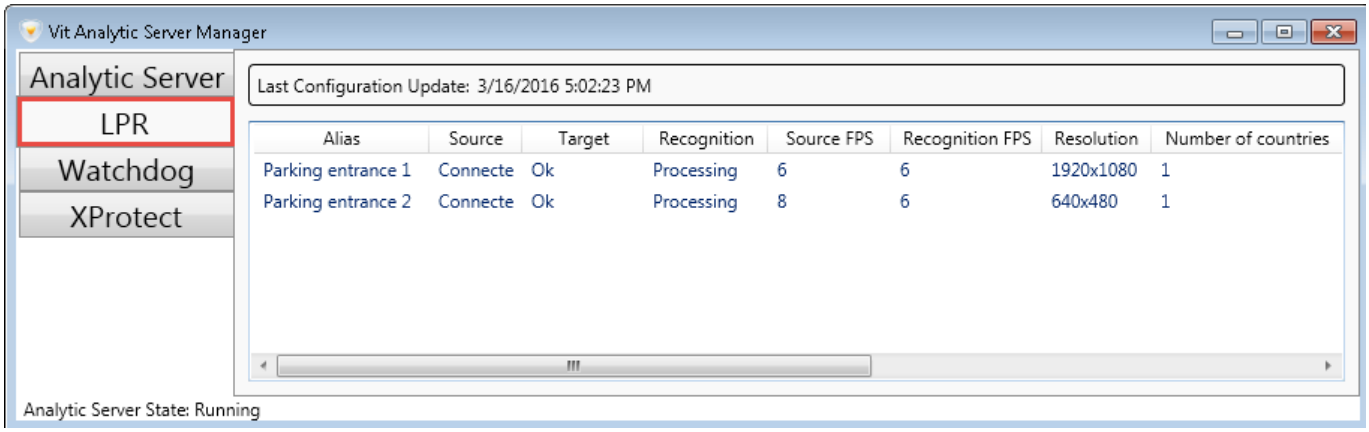


Figure 17: Monitoring feature for active recognition channels, **Analytic Server** manager

- **Name** of a channel.
- **Source** — status of connection to video source. Two possible values for this field are 'Connecting' (establishing connection) and 'Connected' (video is broadcasted).
- **Target** — status of transferring LPR results to a target server. Two possible values for this field are 'Ok' (connection established) and 'Error' (connection failed).
- **Recognition** — status of recognition process on the channel. Two possible values for this field are 'Processing' (recognition in progress) and 'Waiting' (the video is currently not available for processing).
- **Source FPS** — framerate of the video to be processed (frames per second).
- **Recognition FPS** — the number of frames processed by recognition sub-system on this channel per second.
- **Resolution** of frames to be processed.
- **Number of countries** the license plates from which are to be recognized.
- **Frames lost** — the number of frames sent to LPR sub-system, but not processed.

  > The reasons for framerate decimation are described in CPU section.

- **Id** — unique identifier of the channel. System parameter.
- **Error code** — status of the process (for example, 'No errors').

> In case the recognition is not carried out, before contacting technical support it is recommended to perform preliminary troubleshooting. Special checklists were created for this (see the troubleshooting section).

> Pay attention to **Source FPS** and **Recognition FPS** parameters — the values for these parameters should be equal. **Recognition FPS** lower than **Source FPS** means that the CPU capacity of the recognition server is not sufficient enough to process all frames of the video with specified FPS and resolution.

Watchdog tab (Fig. 18) contains information about server resources consumed by recognition processes:

- **Type** of analysis applied to the channel. "VitLpr" stands for license plate recognition.
- **Name** of the recognition channel.
- **PID** — the identifier of the active process.
- **State** of the process:
  - **Stopping** — in process of being disabled (the line is highlighted yellow).
  - **Stopped** — disabled (the line is highlighted red).
  - **Loading** — in process of being enabled (the line is highlighted yellow).
  - **Loaded** — enabled and active (the line is not highlighted).
- **CPU** — processor time consumed by the recognition channel (process), %.
- **RAM** — amount of RAM consumed by the recognition channel (process), Mb.
- **Id** — unique identifier of the channel. System parameter.
- **Errors** — status of a process.
- **Restart** button — disabling and re-enabling the process.

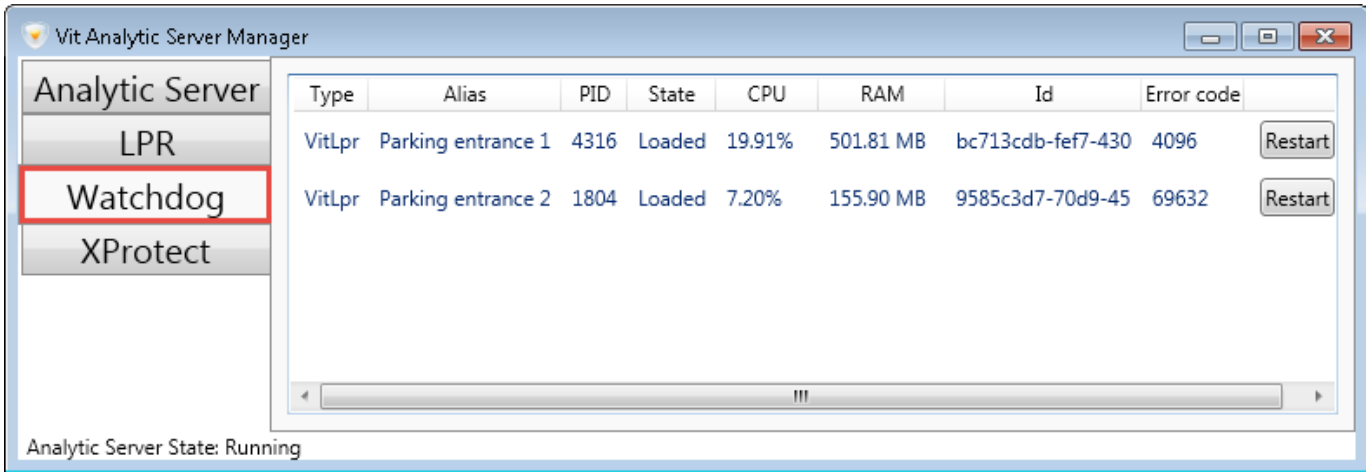


Figure 18: Recognition processes control feature, **Analytic Server** manager

Configuring Logging Parameters

To configure logging in LPR sub-system (Fig. 19):

1. Open **VIT Analytic Server** manager.
2. Switch to **Analytic Server** tab.
3. Specify the following parameters:
   - **Configuration** for receiving messages from third-party systems. Recommended value is 'Normal'. Debugging mode is to be used by developers and support engineers.

- **Log types**:
  - **Debug** — recording additional set of parameters that are unique to each channel (debug messages) to the log file (to be used by developers).
  - **Informational** — recording information about all the events from all modules to the log file (info messages).
  - **Warning** — recording alert messages to the log file (warning messages).
  - **Error** — recording errors in module operation to the log file (error messages).
- **Log File Size** — maximum size of the log file that can be recorded by the system (maximum value is 1024 Mb).

4. Click **Save and Apply** button (or click **Cancel** to discard changes).

The log file may be accessed by clicking **View log File** button (default directory — C:\ProgramData\VIT\Analytic Server\logs). This file may be requested by technical support engineers in case of any issues with **AutoCode XProtect** performance.
Stop Analytic Server button disables the service (**vitanalyticsrv**).

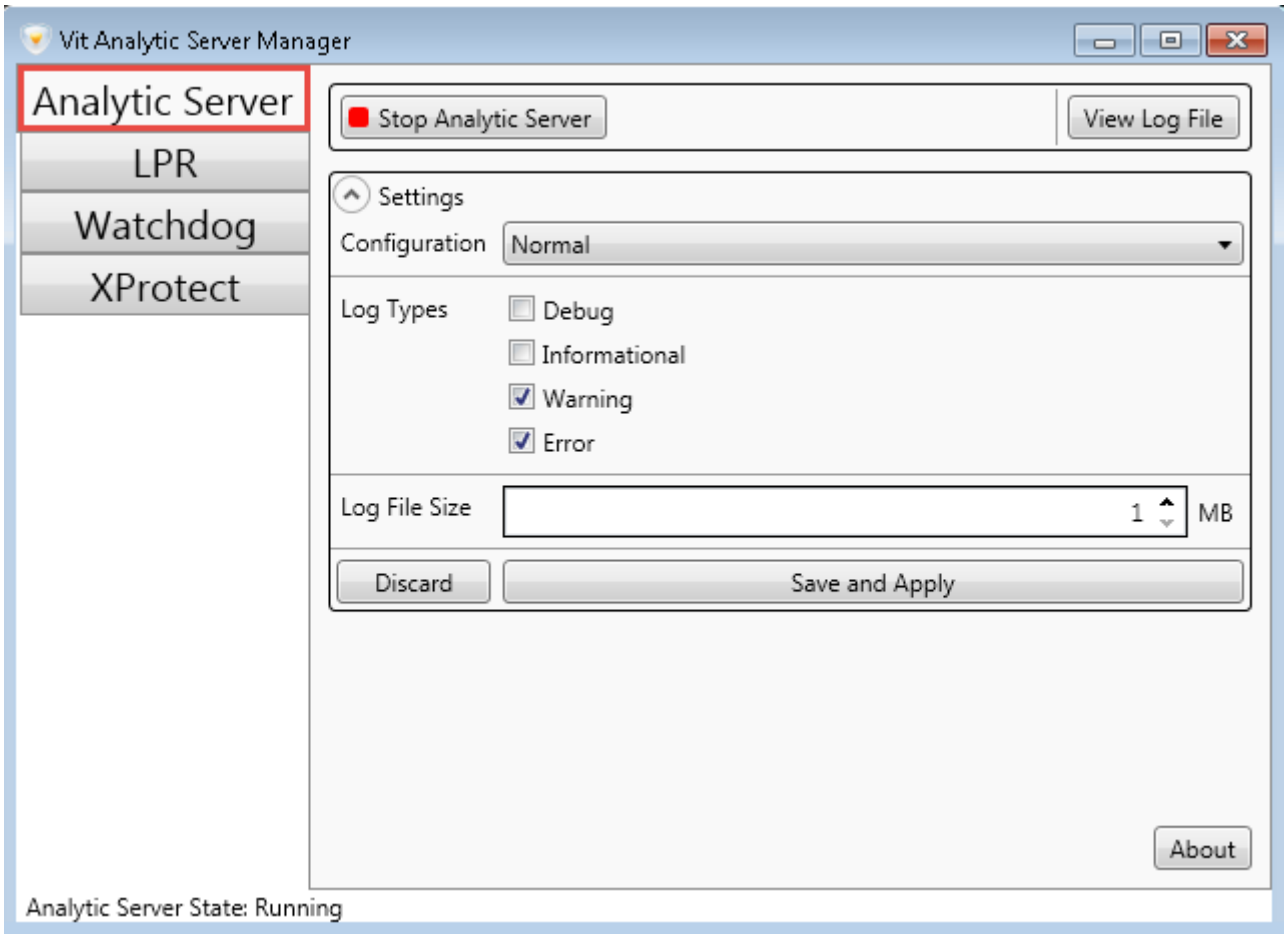
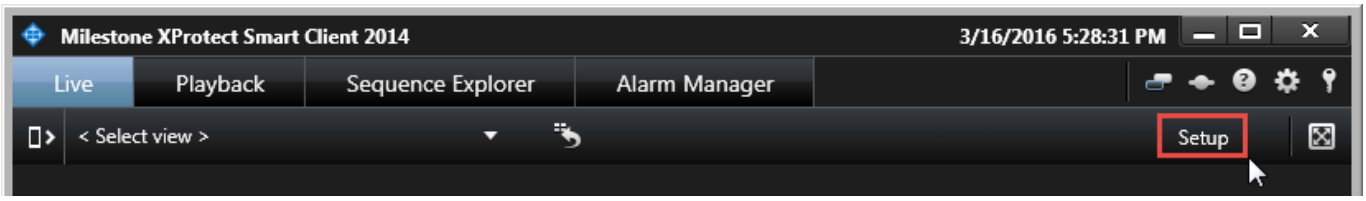


Figure 19: Logging feature of LPR sub-system, **Analytic Server** manager

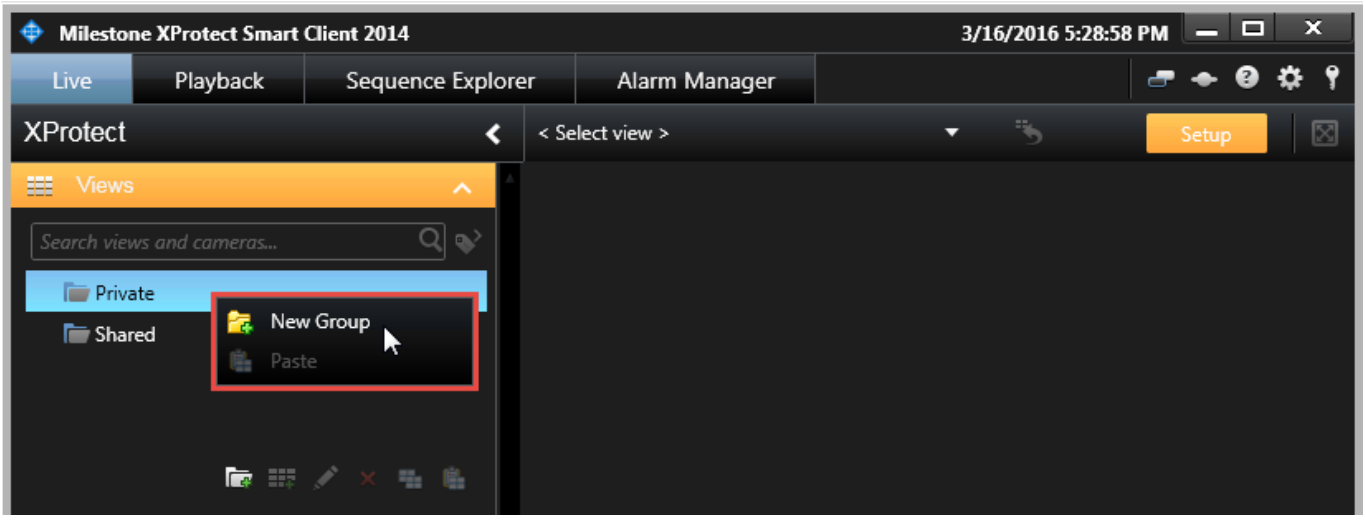
6 CONFIGURING THE VIEWER MODULE

The license plate protocol in **XProtect Smart Client** may be configured on a computer with **Viewer** module pre-installed. To display events in client application:

1. Launch **XProtect Smart Client** and log in.
2. Click **Setup** button to access configuration.

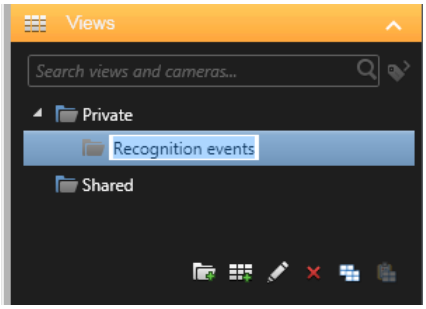


3. Select either **Private** or **Shared** folder on **Views** panel. Open context menu and select **New Group**.



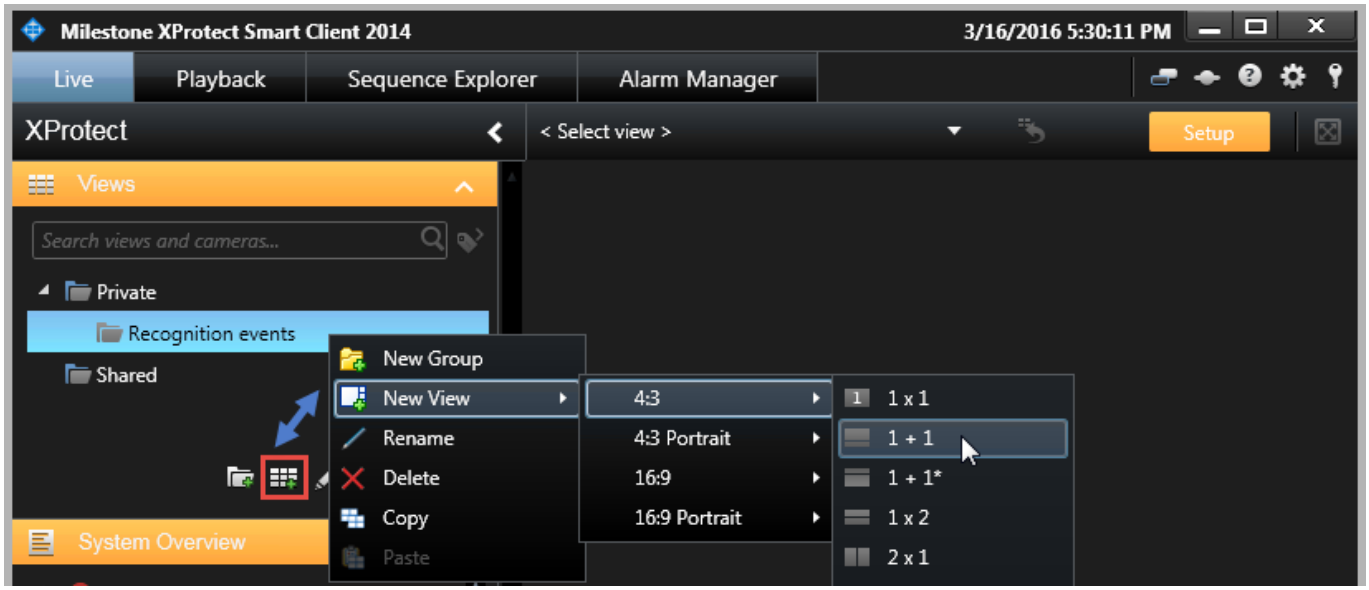
> Selecting the type of view (**Private**/**Shared**) depends on its purpose and user rights. Private views are only available to the user that created them. Shared views are simultaneously available to multiple **XProtect Smart Client** users. Depending on the kind of surveillance system, shared views may be available for all users or a specific group of them. A detailed description of views creation and distribution of rights to edit them may be found in **XProtect Smart Client** User Guide.

4. Name the new views folder.

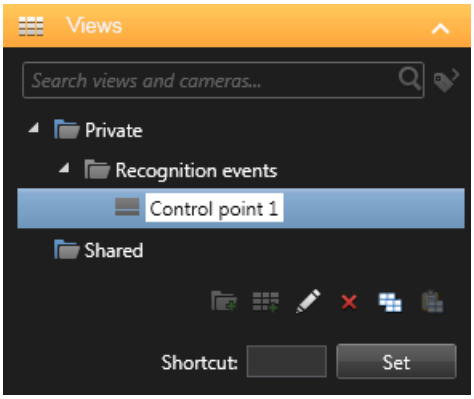


5. Select **New View** option from the context menu of new folder (or select the folder and click the button to add new view at the bottom of the panel).

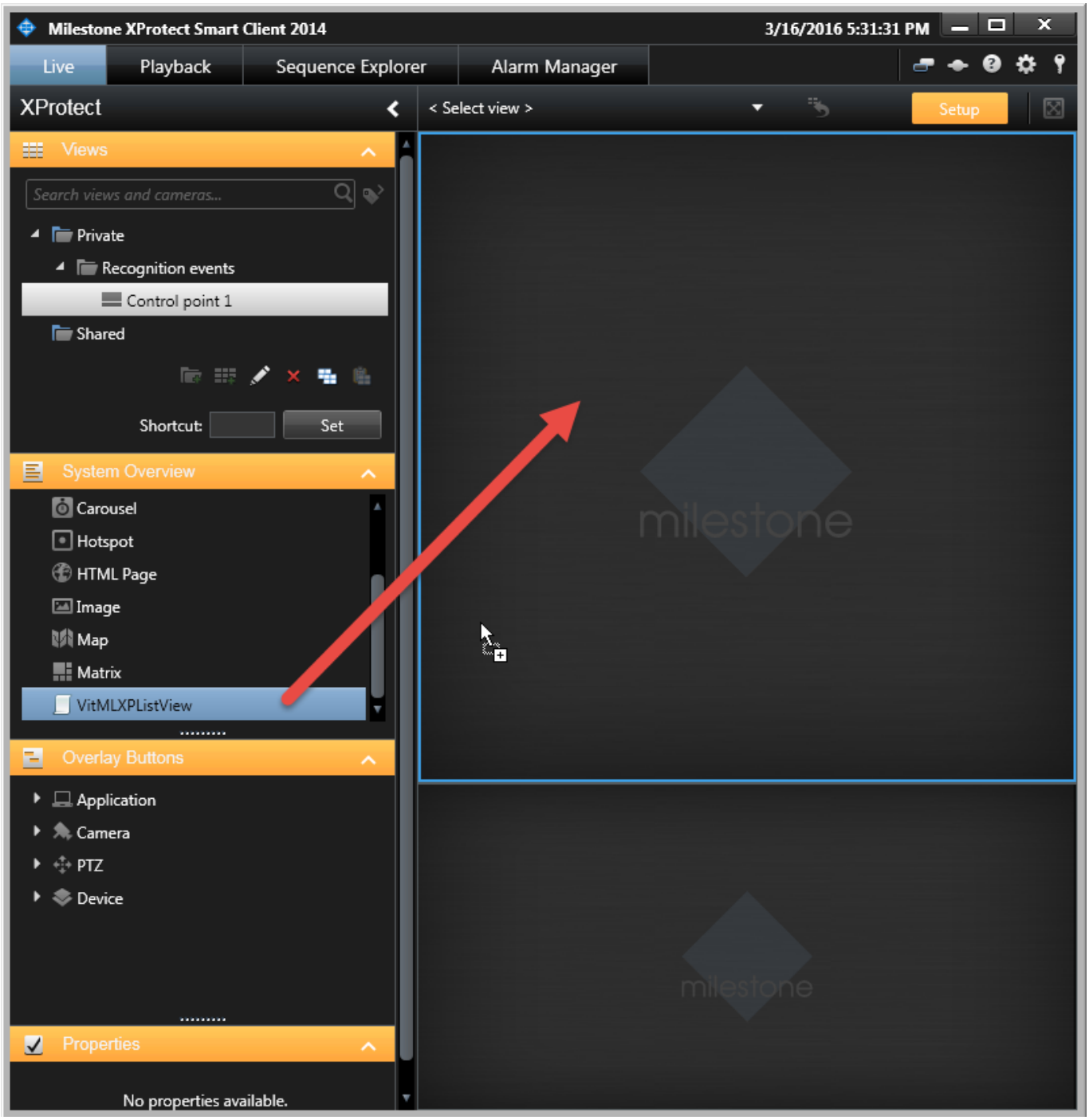
6. Select the layout for a new view.



7. Name the new view.



8. Select **VitMLXPListView** element (**Viewer** module graphic interface) on **System Overview** panel and drag it into view.



9. Select the type of messages to be shown in license plate protocol (**Properties** panel, **Data Source** parameter):

   - **Alarms** — all alarms within **XProtect** system including those generated based upon analytic events of **LPR** module — **Plate detected**, **Car detected**, **Car lost**.
   - **Events** — all analytic events within **XProtect** system including those generated by **LPR** and **Gateway** modules (see the List of AutoCode XProtect analytics events annex).

   > Internal messages would not be displayed in **XProtect** license plate protocol.

10. Set the maximum amount of lines to be shown on one page of the recognition protocol (set to 100 by default).

The configuration should result in recognition events being displayed in **XProtect Smart Client** interface (Fig. 20).

AutoUpdate < 1 - 100 > Create Export Filter

Message	Source Name	Object Value	Timestamp	Tag	LocalId	Vendor Name	Post	Speed
Car lost	Camera 1	KDO46	15:44:06 2/6/2016	0	17823	"Video Internet Tec...		
Plate detected	Camera 1	KDO46	15:44:06 2/6/2016	0	17822	"Video Internet Tec...		
Car detected	Camera 1	KDO46	15:44:06 2/6/2016	0	17821	"Video Internet Tec...		
Car lost	Camera 2	SHM551	15:44:05 2/6/2016	0	17817	"Video Internet Tec...		
Car lost	Camera 2	BZH455	15:44:04 2/6/2016	0	17814	"Video Internet Tec...		
Car detected	Camera 2	BZH455	15:44:04 2/6/2016	0	17813	"Video Internet Tec...		
Plate detected	Camera 2	BZH455	15:44:04 2/6/2016	0	17812	"Video Internet Tec...		
Car lost	Camera 1	ESO26	15:44:04 2/6/2016	0	17820	"Video Internet Tec...		
Car detected	Camera 1	ESO26	15:44:04 2/6/2016	0	17819	"Video Internet Tec...		
Plate detected	Camera 1	ESO26	15:44:04 2/6/2016	0	17818	"Video Internet Tec...		
Car lost	Camera 2	BZH455	15:44:01 2/6/2016	0	17808	"Video Internet Tec...		
Plate detected	Camera 2	BZH455	15:44:00 2/6/2016	0	17807	"Video Internet Tec...		
Car detected	Camera 2	BZH455	15:44:00 2/6/2016	0	17806	"Video Internet Tec...		
Car lost	Camera 1	DYO659	15:44:00 2/6/2016	0	17811	"Video Internet Tec...		
Plate detected	Camera 2	SHM551	15:43:57 2/6/2016	0	17815	"Video Internet Tec...		
Plate detected	Camera 1	DYO659	15:43:56 2/6/2016	0	17810	"Video Internet Tec...		
Car lost	Camera 2	BIR038	15:43:54 2/6/2016	0	17805	"Video Internet Tec...		
Plate detected	Camera 2	BIR038	15:43:53 2/6/2016	0	17803	"Video Internet Tec...		
Car detected	Camera 2	SHM551	15:43:51 2/6/2016	0	17816	"Video Internet Tec...		
Car detected	Camera 2	BIR038	15:43:48 2/6/2016	0	17804	"Video Internet Tec...		
Car lost	Camera 2	BRH317	15:43:46 2/6/2016	0	17802	"Video Internet Tec...		

Figure 20: Interface of the recognition protocol after being added to **XProtect Smart Client** view

Working with license plate protocol in **XProtect Smart Client** includes viewing and managing recognition events. To achieve this, filtering, sorting, editing, importing and exporting features were introduced. Instructions for working with **Viewer** module may be found in **AutoCode XProtect** operator's guide.

7 GATEWAY MODULE OPERATION

module overview

Gateway module is an **XProtect event server** plug-in configured using **XProtect Management Client** UI. The module is capable of:

- filtering specific events generated in **XProtect**;
- creating event processing logic using scripts or handlers (provided with the product or created independently);
- communicating with external databases;
- creating unique events and sending them to the **XProtect** event server;
- communicating with other applications on SOAP level.

Capabilities of this module go beyond LPR-related tasks. However, in this document they are illustrated using the following use cases:

- Use case 1: Calculating time spent at parking
- Use case 2: Searching for recognized license plates in the list

This allows describing features of **Gateway** module in practical environment.
In **Management Client** UI, **Gateway** module features are presented as system objects grouped under **VitM-LXP.Gateway** parent object within **MIP Plug-ins**.

scripts

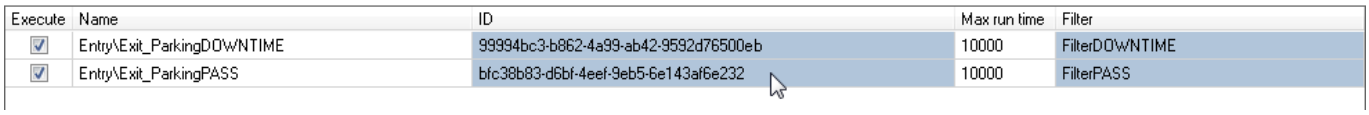
In the context of **Gateway** module, a script is a source code used to specify reactions to events (either generated within **XProtect** or imported).
IronPython 2.7.4 is used which is one of main versions of Python programming language designed for Microsoft .NET platform. Therefore, aside from standard Python capabilities, it is possible to use .NET Framework types directly in scripts.
A script may be integrated into **XProtect** in following ways:

- by being developed manually;
- by importing supplied scripts, i.e. scripts created by **AutoCode XProtect** developers for the user based on specific requests (importing is described in Use case 1: Calculating time spent at parking).

To create a script manually, the **Milestone Integration Platform** (MIP SDK) software development package is required. Pre-made **Gateway** module scripts that can perform most common tasks required by the user may be provided based on the requirements of the client.

> **Gateway** module also includes a processing feature, represented by 'handler' files. They may be used to execute protected scripts saved as .dll files.
> Handlers may be used as an alternative to scripts. This document only describes scripts usage.

Script editor is opened in a new window upon double-clicking a respective **Script** object (Fig. 21).



Execute	Name	ID	Max run time	Filter
☑	Entry\Exit_ParkingDOWNTIME	99994bc3-b862-4a99-ab42-9592d76500eb	10000	FilterDOWNTIME
☑	Entry\Exit_ParkingPASS	bfc38b83-d6bf-4eef-9eb5-6e143af6e232	10000	FilterPASS

Figure 21: Properties panel of a **Script** object

A **Script** window appears that contains a toolbar in the top section. The toolbar is active if **Main part** or **Init part** tabs are activated. A description for each element is shown if the cursor is hovered over it (Fig. 22).

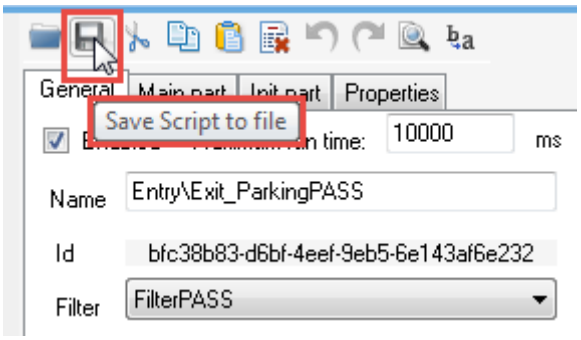
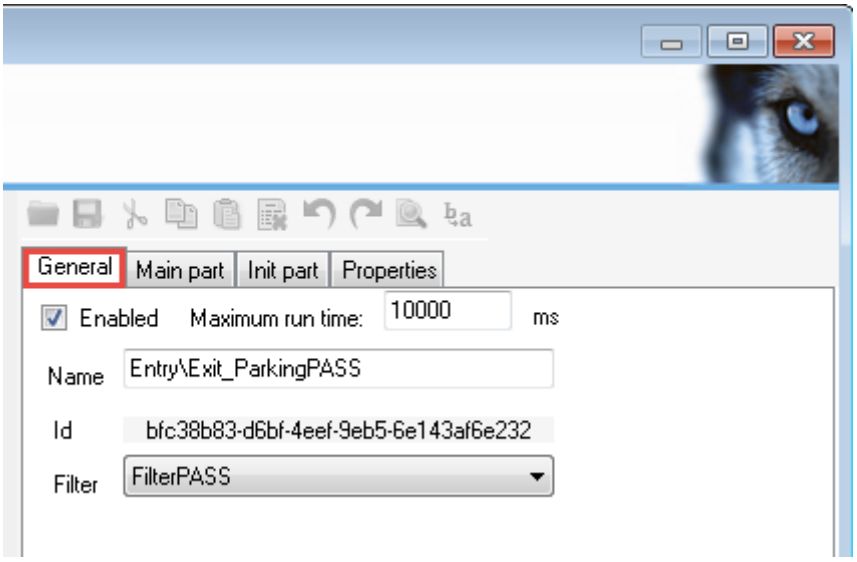


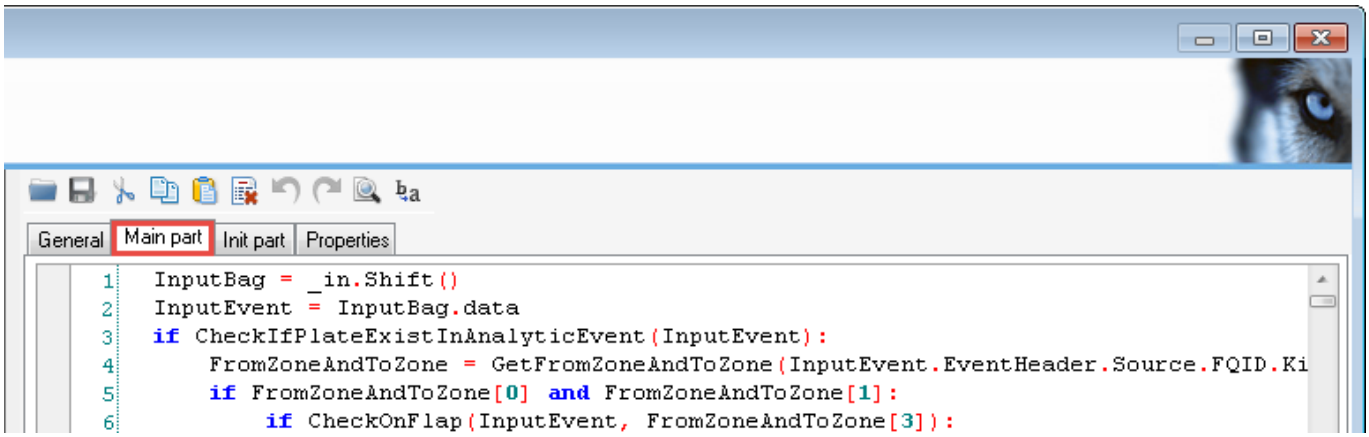
Figure 22: The toolbar inside **Script** object editing window

1. The **General** tab contains general information about the script:

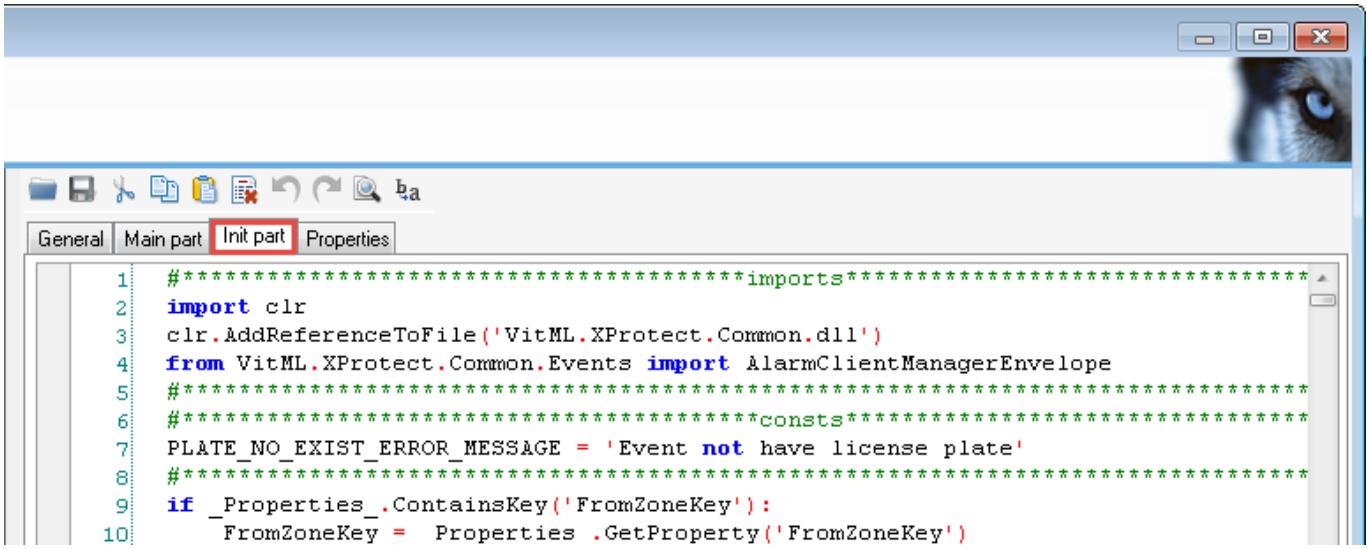
   - **Enabled** — a checkbox to activate the script.
   - **Name** of the script.
   - **ID** — an identifier of the script. This is an internal parameter not available for editing.
   - **Filter** — an object with the same name created within **Gateway** module. It performs the function of selecting specific events among those available to the module to be used by the script.
   - **Maximum run time** — maximum period of time for the script to be executed. Should the time limit be exceeded, the execution of the script is aborted.



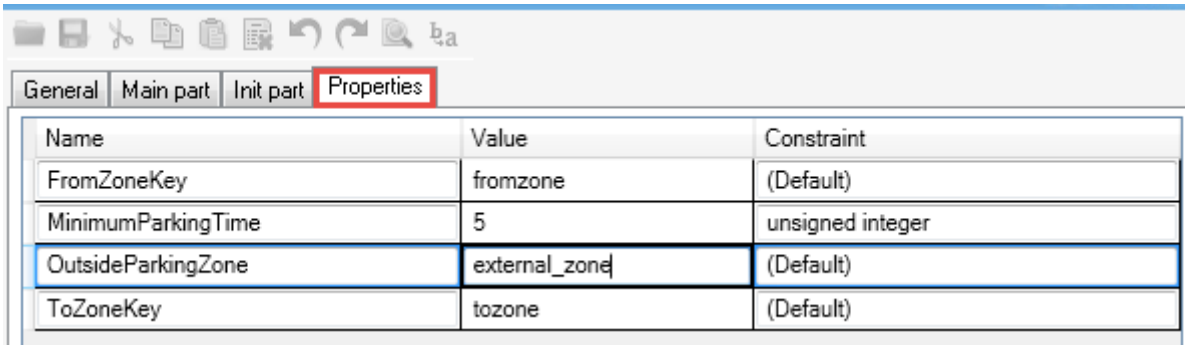
2. The **Main part** tab contains the scenario for reacting to specific events.



3. The **Init part** tab allows connecting to third-party modules. It contains the features and parameters used in main section.



4. The **Parameters** tab allows configuring key user parameters without manually editing the script.



use case 1: calculating time spent at parking

Purpose: recording the time of vehicles entering the parking lot; recording the time of vehicles leaving the parking lot identifying them by license plates recognized when they entered; calculating the time vehicles spent at the parking lot, for example, to create statistics or calculate the cost of services provided.

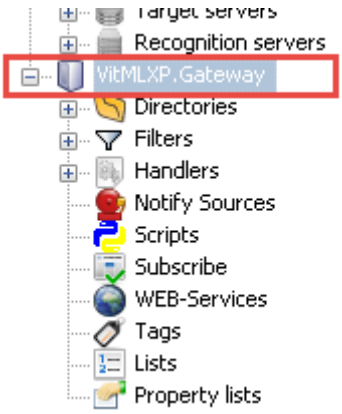
Stages of solution:

1. Setting up recognition on videostreams from the cameras installed at the parking (see previous sections of this document).
2. Creating event processing logic:
   - Importing scripts
   - Configuring scripts
   - Determining special events
   - Creating a property list
3. Processing results.

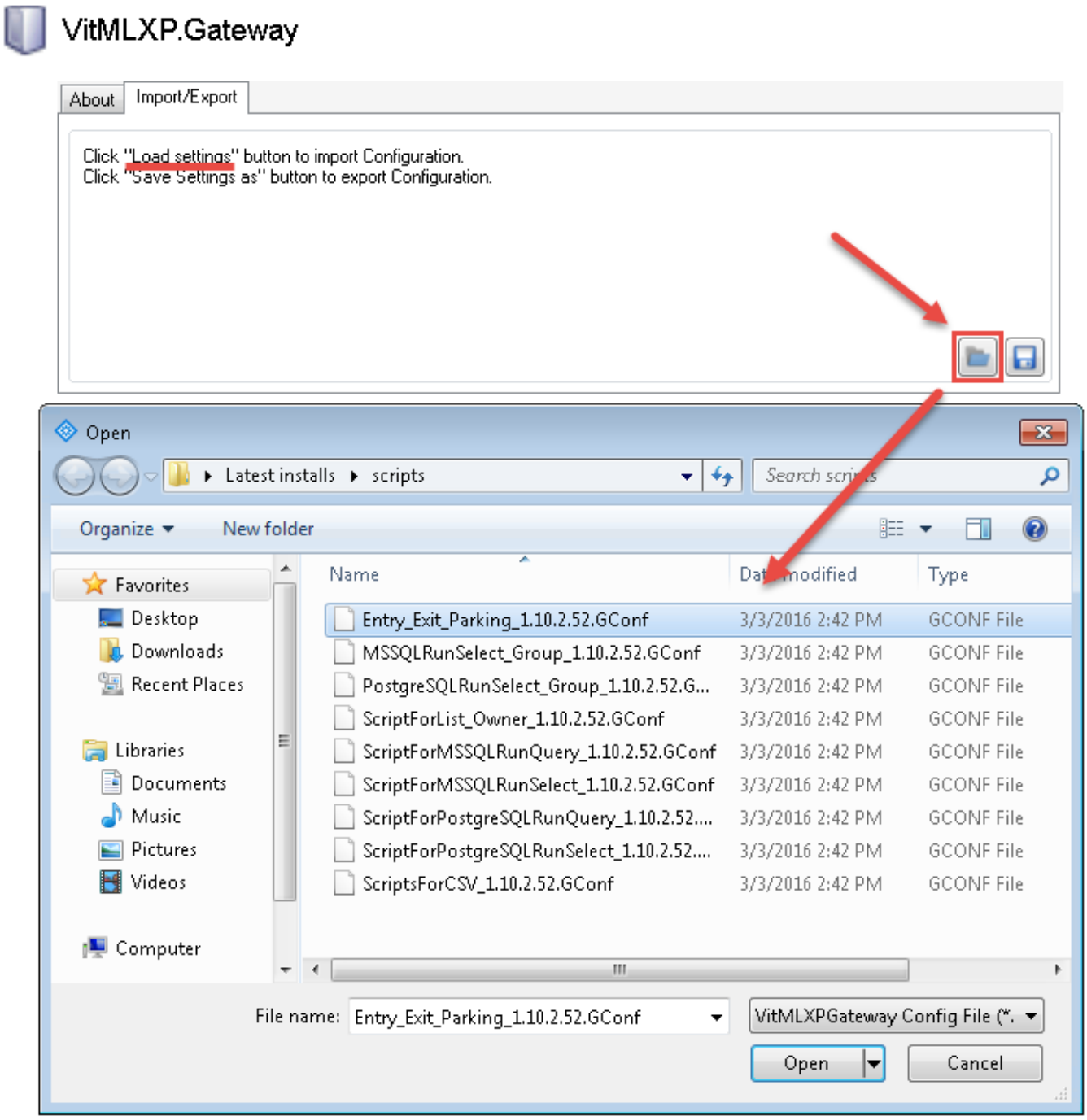
Importing scripts

To import a script (for calculating time spent on parking) into **XProtect** system:

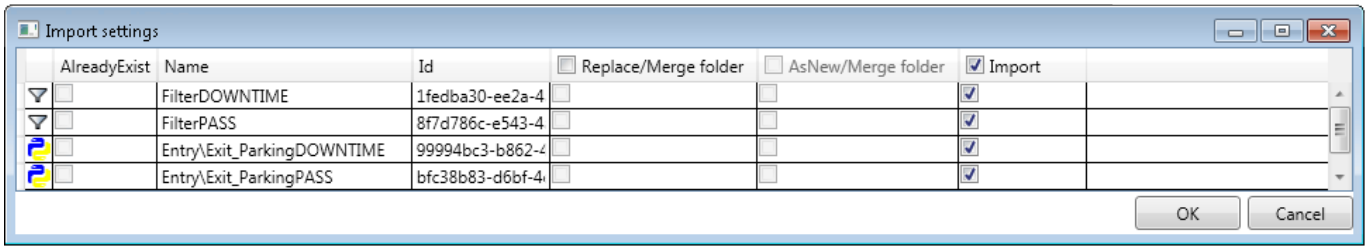
1. Select **VitMLXP.Gateway** parent object.



2. Select **Import/Export** tab in the configuration tool that appears on the right.
3. Click the **Load Settings** button.
4. In the window that appeared, specify the path to a **Entry_Exit_Parking.GConf** configuration file.
5. Import the file using **Open** button.



6. See the list of functionalities being imported in the **Import Settings** window. The fields in the table distinguish between the files imported for the first time (**Import** column) and already existing files (**AlreadyExist** and **Replace/Merge** folder columns). Click **OK**.



Import result: when the **Script** object is selected, the following parameters are to be shown:

- **Entry/Exit_ParkingPASS** — script to record the time the vehicle arrived/left.

- **Entry/Exit_ParkingDOWNTIME** — script to calculate the time spent on parking.

Scripts:

Execute	Name	ID	Max run time	Filter
☑	Entry\Exit_ParkingDOWNTIME	99994bc3-b862-4a99-ab42-9592d76500eb	10000	FilterDOWNTIME
☑	Entry\Exit_ParkingPASS	bfc38b83-d6bf-4eef-9eb5-6e143af6e232	10000	FilterPASS

Figure 23: Configuring the **Entry/Exit_ParkingPASS** script

Configuring scripts

Entry/Exit_ParkingPASS script

To record entering/exiting:

1. Open **Entry/Exit_ParkingPASS** script editor.
2. Switch to **General** tab. Select **FilterPASS** from **Filter** dropdown. Make sure that **Enabled** checkbox is selected.
3. Switch to **Parameters** tab and set the following parameters of script operation in respective fields of **Value** column (Fig. 24):
   (a) **ToZoneKey** is an additional parameter for a **Camera** object, created using property list feature (see Creating a property list). The value of **ToZoneKey** parameter would be used to state that the camera is recording the fact of a vehicle *entering* the parking zone. Value format: string. Example value: 'tozone'.
   (b) **FromZoneKey** is an additional parameter for a **Camera** object, created using property list feature. The value of **FromZoneKey** parameter would be used to state that the camera is recording the fact of a vehicle *leaving* the parking zone. Value format: string. Example value: 'fromzone'.
   (c) **OutsideParkingZone** is a name used to denominate the territory outside the parking lot. Value format: string. Default value: '0'.
   (d) **MinimumParkingTime** is a minimum allowed time the vehicle can spend on the parking lot in seconds. Value format: whole number. Default value: '5'.

General | Main part | Init part | Properties

Name	Value	Constraint
FromZoneKey	fromzone	(Default)
MinimumParkingTime	5	unsigned integer
OutsideParkingZone	external_zone	(Default)
ToZoneKey	tozone	(Default)

Figure 24: Configuring the **Entry/Exit_ParkingPASS** script

Entry/Exit_ParkingDOWNTIME script

To calculate time spent on parking:

1. Open **Entry/Exit_ParkingDOWNTIME** script editor.
2. Switch to **General** tab. Select **FilterDOWNTIME** from **Filter** dropdown. Make sure that **Enabled** checkbox is selected.
3. Switch to **Parameters** tab and set the following parameters of script operation (Fig. 25):
   - **MaximumParkingTime** — maximum period of time a single vehicle may remain on parking in hours. Default value: 120 (whole number). Entrance events would be searched for within that period of time.
   - **OutsideParkingZone** is a name used to denominate the territory outside the parking lot. Default value: '0' (string).

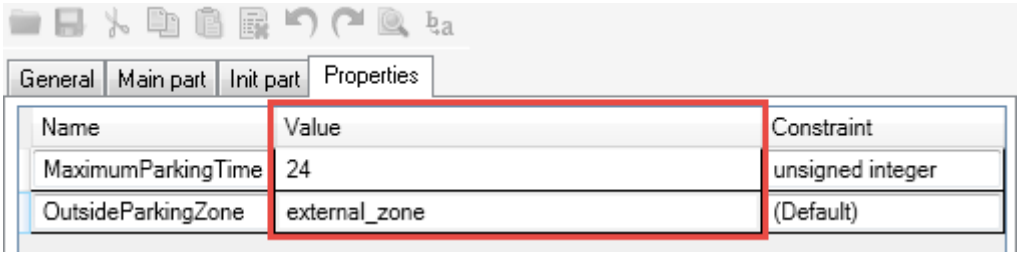
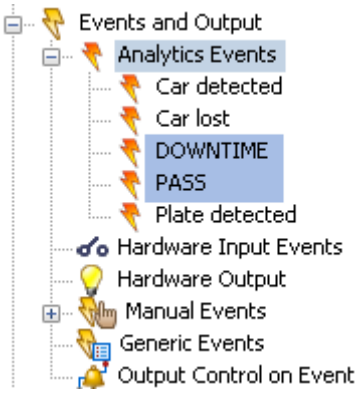


Figure 25: Configuring the **Entry/Exit_ParkingDOWNTIME** user script

Determining special events

The following recognition events are to be created:

- **PASS** — the event is generated when vehicle passing time is registered.
- **DOWNTIME** — the event is generated after the time the vehicle spent on parking is calculated (it contains the name of the area as well as entrance/exit time).



> The process of creating new analytic events with examples of **Plate detected**, **Car detected**, **Car lost** is described in Registering Analytics Events section.

Creating a property list

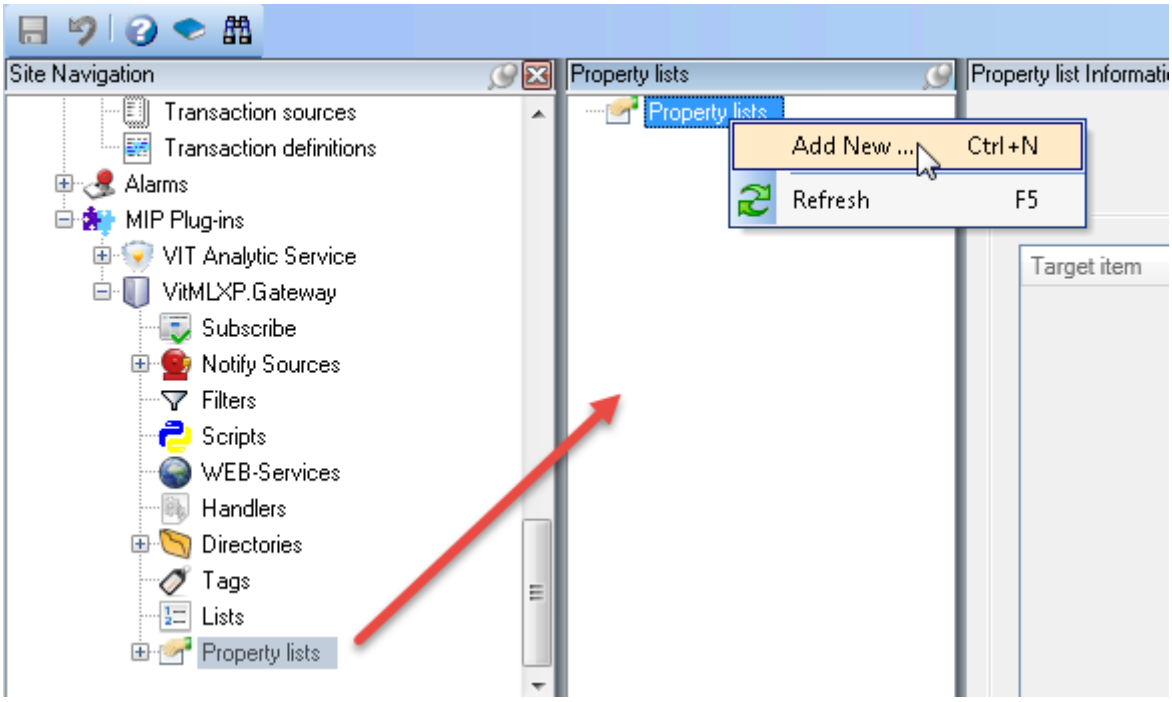
When the processing scripts are configured, it is necessary to specify the cameras to register vehicles entering parking as well as cameras to register them leaving. This task can be solved using **property list** feature.

Additional properties are assigned to active surveillance cameras to determine whether the vehicle is entering parking area or exiting it.
This allows for control over the movement of the vehicle on the parking lot.

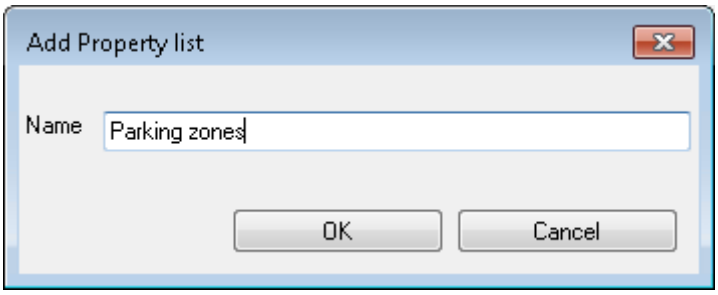
Creating instance

1. Select a **Property lists** object on navigation panel.

2. Open context menu of an object with the same name on overview panel and select **Add New** option.



3. Name the new sample (e.g. 'Parking zones') and click ".



4. Save the configuration.

Configuring instance

After that open the properties panel of the new sample and click **Add** button to create necessary amount of **Key/Value** lines. Each line represents a single new parameter for the **Camera** object (Fig. 26).
Minimum required amount of lines for a single property list is 4:

- to register entrance/exit at a parking that is considered to be one zone, two cameras must be used.
- two parameters are to be set for each camera:
  - name of the zone, exit *from* which is being registered

– name of the zone, entrance *to* which is being registered

List name: Parking zones

Content

Target item	Key	Value
Camera 1 [All Cameras (Server)]	fromzone	external_zone...
Camera 1 [All Cameras (Server)]	tozone	parking...
Camera 5 [All Cameras (Server)]	fromzone	parking...
Camera 5 [All Cameras (Server)]	tozone	external_zone...

Figure 26: Example of a property list for registering entrance/exit at a parking

Parking areas may be divided into sub-zones in which case the number of cameras is increased accordingly to register entrance/exit between sub-zones and, therefore, the number of properties in the list is increased as well.

The following parameters are to be set for each line (Fig. 26):

1. **Target element** — name of the camera registering entrance/exit of the vehicle.
2. **Key** — one of the values set for **FromZoneKey** or **ToZoneKey** parameters. This value would determine whether the camera register the vehicle entering or exiting the zone.
3. **Value** — name of the zone the vehicle is leaving (if the exit key is set) or entering (if the entrance key is set).

For example, (Fig. 27):

- Two devices are used to record vehicles entering/exiting the zone. The devices are identified within **XProtect** as **Camera 1** and **Camera 5**.
- "fromzone" and "tozone" values are set for **FromZoneKey** and **ToZoneKey** parameters respectively in **Entry/Exit_ParkingPASS** script.
- "external_zone" value is set for **OutsideParkingZone** parameter in **Entry/Exit_ParkingDOWNTIME** script.
- parking name was assigned to the area within parking (set as **Value** parameter in property list).

Keep in mind that name of the zone outside parking must be the same as the value inserted in **OutsideParkingZone** parameter of **Entry/Exit_ParkingDOWNTIME** script.

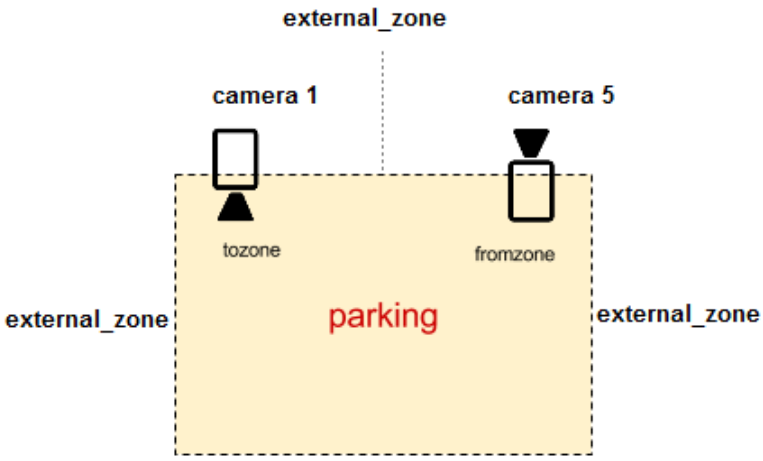


Figure 27: Visual representation of **Entry/Exit_ParkingPASS** script parameters

Processing results

The result of importing and configuring special scripts to calculate time vehicles spent at parking should be **PASS** and **DOWNTIME** events displayed in **XProtect Smart Client** (see description of events in Determining special events).
Aside from **General** parameters such as **Timestamp** and **ObjectValue**, the **DOWNTIME** event would have the following additional parameters:

- **Zone** — name of the parking zone the vehicle stopped in.
- **Time in** — time of license plate recognition at the moment the vehicle enters the parking zone.
- **Time out** — time of license plate recognition at the moment the vehicle exits the parking zone.
- **Park** — time difference between entrance and exit of the vehicle.

> Setting up recognition views is described in configuring the viewer module section. Parameters of events to be displayed in license plates protocol (**Viewer** module graphic interface) are described in **AutoCode XProtect** user manual.

> In order to have **PASS** and **DOWNTIME** events displayed in **Alarm manager**, an **Alarm event** should be configured in **Management Client**. The **Trigger event** parameter should be set to:
>
> - **PASS** if the alarm should be created when the vehicle passes the camera's control zone.
> - **DOWNTIME** if the alarm should be created when the time at parking is calculated.

The following events are available additionally:

- **vitml.error.acs.pass_flap** — time the vehicle spent in the zone does not reach minimum allowed time (a duplicate recognition event is possible).
- **vitml.error.acs.port_incorrect** — event of entering the zone that was not defined in property list settings.
- **vitml.error.acs.lost_inpass** — event of registering the exit of a vehicle that was not registered to enter the zone.

- **vitml.error.acs.lost_outpass** — event of the time the vehicle spent in the zone exceeding the maximum allowed time period (entrance registered, exit not registered).

use case 2: searching for recognized license plates in the list

Purpose: when the license plate is recognized, search for it in specified lists; generate an event based on search results.

Ways of solution:

1. search for the license plate in the list displayed as **XProtect** system object (hereinafter — local list);
2. search for the license plate in external storages (**Microsoft SQL Server** or **PostreSQL**) not connected to **XProtect** database.

Stages of solution:

1. Configuring LPR — basic stage with no regards of where the list is located.
2. Working with the source for comparison:
   - *For a local list:*
     - creating one or multiple **List** objects (manually or using a CSV import); if the list is created manually, it must be filled with data to compare recognized license plates;
     - creating a **Tag** object to logically connect lists to the source of recognition events (camera);
     - determining special events;
     - importing either **ScriptForList_Owner** or **ScriptForCSV** based on the contents of the list.

       > Detailed instructions may be found in Local list sub-section.
   - *For an external depository:*
     - importing one of special scripts;
     - configuring connection to the depository.

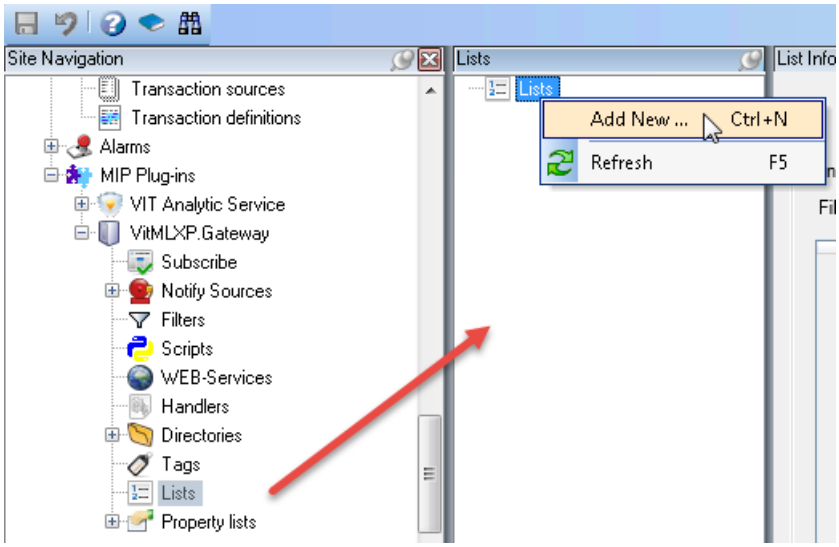
       > Detailed instructions may be found in External database sub-section.

Local list

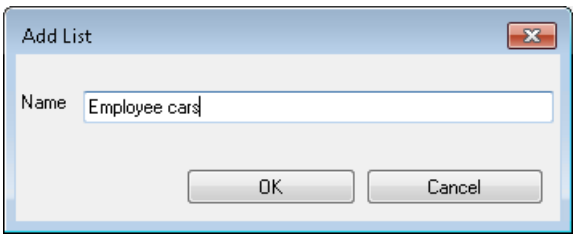
Step 1. Creating the list

The purpose of a **List** object is to store user data (including license plates) within **XProtect** system.
To create an instance of a **List** object:

1. Select a **Lists** object on a navigation panel.
2. Open context menu of the object and click **Add new** on the overview panel located to the right.



3. Name the new list in the window that appears (e.g. "Employee cars").



4. Save the configuration.

When the list is created, it is to be filled with data to be compared to license plates after recognition. It may be done manually (Creating the list manually section) or by importing a CSV-file (Importing the list using a CSV file section).

> Data in the list is recorded as table. The **requirement** that the data must meet whether the list is imported or created manually is to have a PLATE column. This column would contain license plates to be compared to recognition results.

creating the list manually

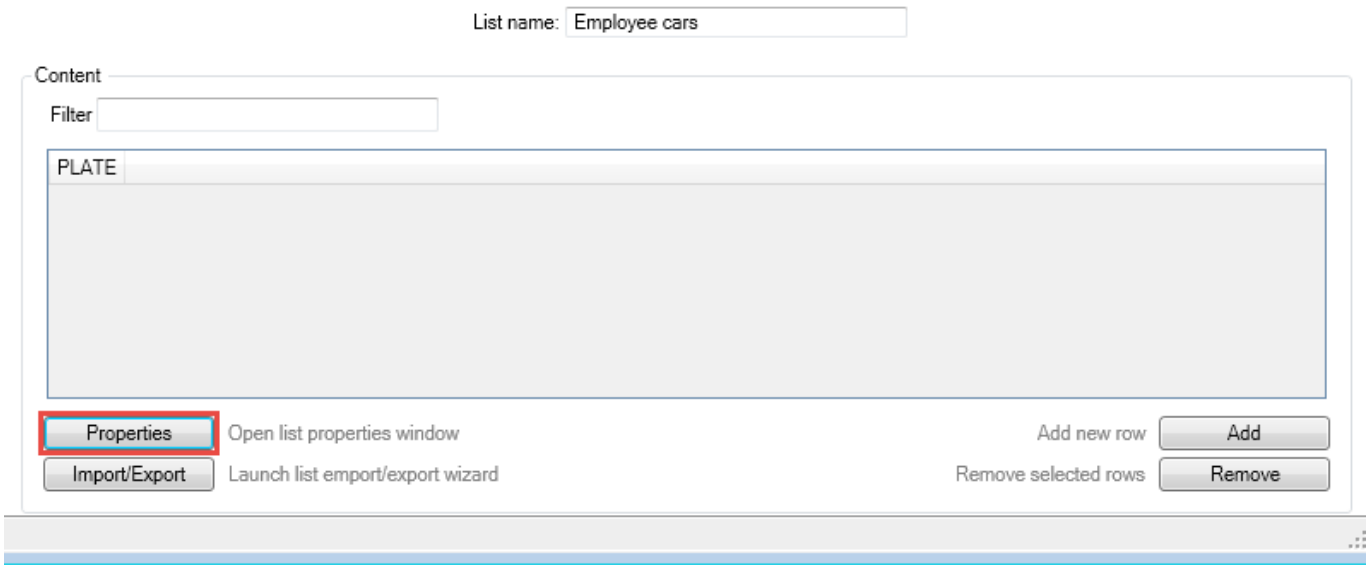
1. Open properties of the newly created **List** object instance.

2. Click the guielementProperties button to add the mandatory PLATE column as well as additional columns (e.g. **Name** column to store names of car owners). If PLATE is the only column necessary, skip Step 3 and proceed from Step 4 for if these instructions.

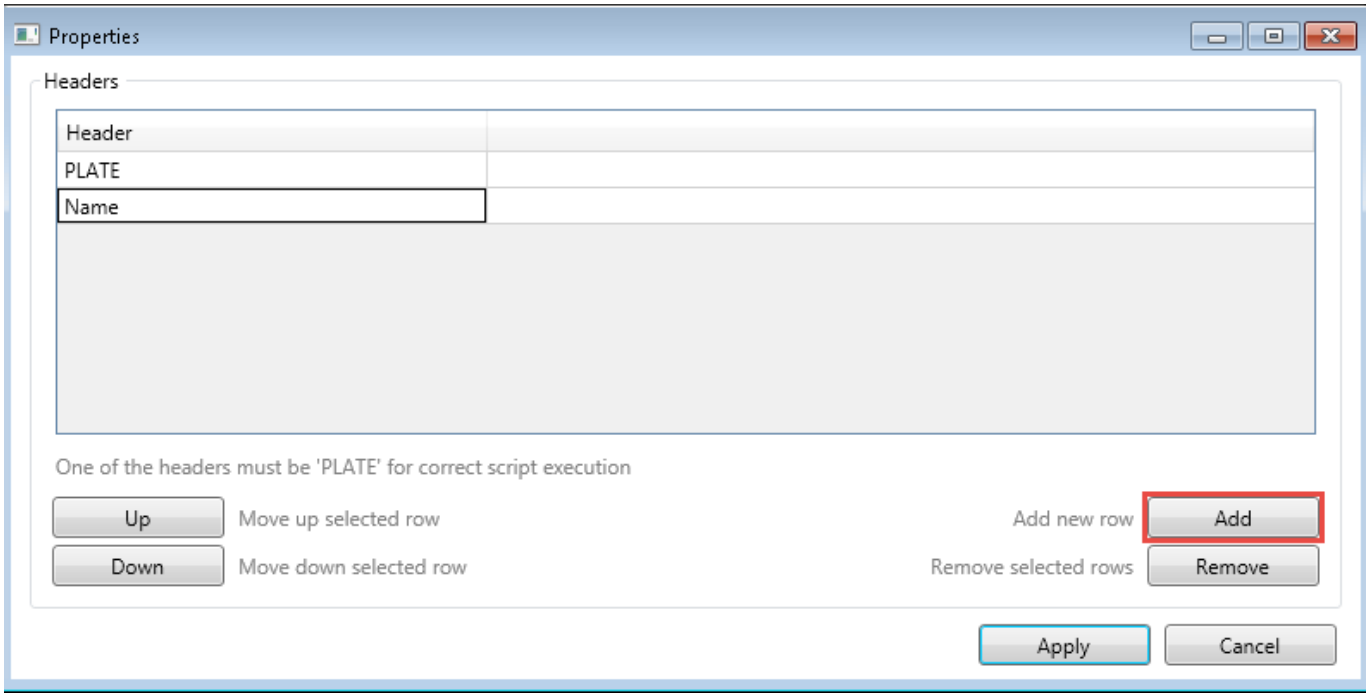
   > Adding new columns is only possible before the data is inserted in the table. After at least one line of data is added, creating additional columns is not possible.

   > Step 4 of these instructions requires either **ScriptForCSV** or **ScriptForList_Owner** to be imported. **ScriptForCSV** is used if PLATE is the only mandatory column in the table. **ScriptForList_Owner** is used if the list is supposed to contain information about owners of the vehicles the license plates of which are stored in PLATE column. In latter case the names of owners are to be recorded in **Owner**

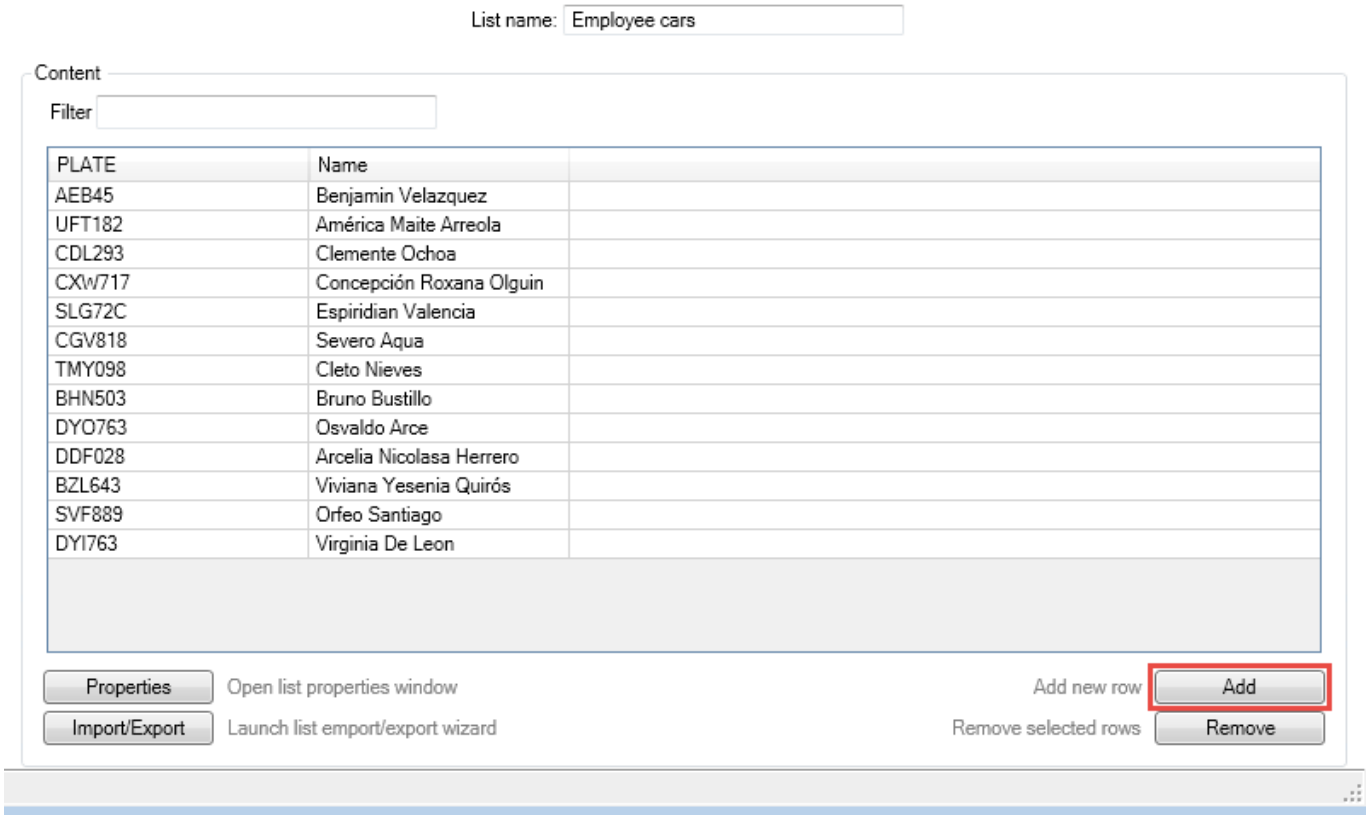
> column. These instructions do not describe this scenario and as an example names of owners would be recorded in **Name** column (with **ScriptForCSV** script used, respectively).



3. Click **Add** button on the **Properties** window that appeared to add a column to the list. Name the column (**Title** parameter). Use **Up/Down** buttons to adjust the position of the highlighted column (displayed as a line). Click **Apply** after the columns are created.

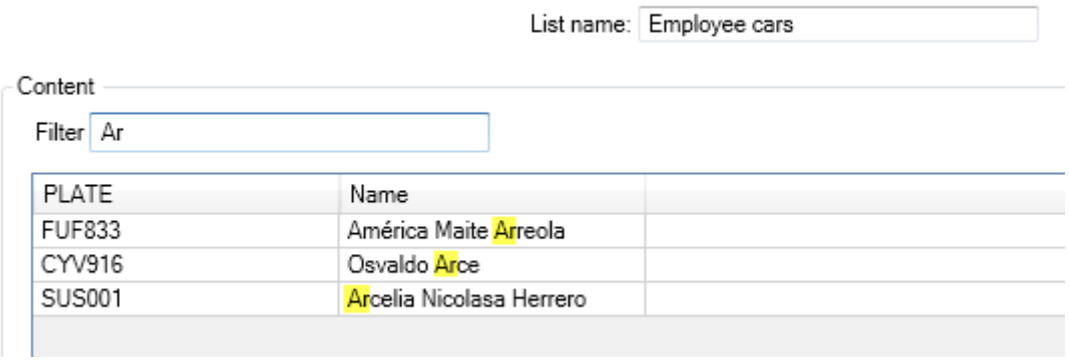


4. Fill the list with data. Click **Add** button to add a new line.



5. Save the configuration.

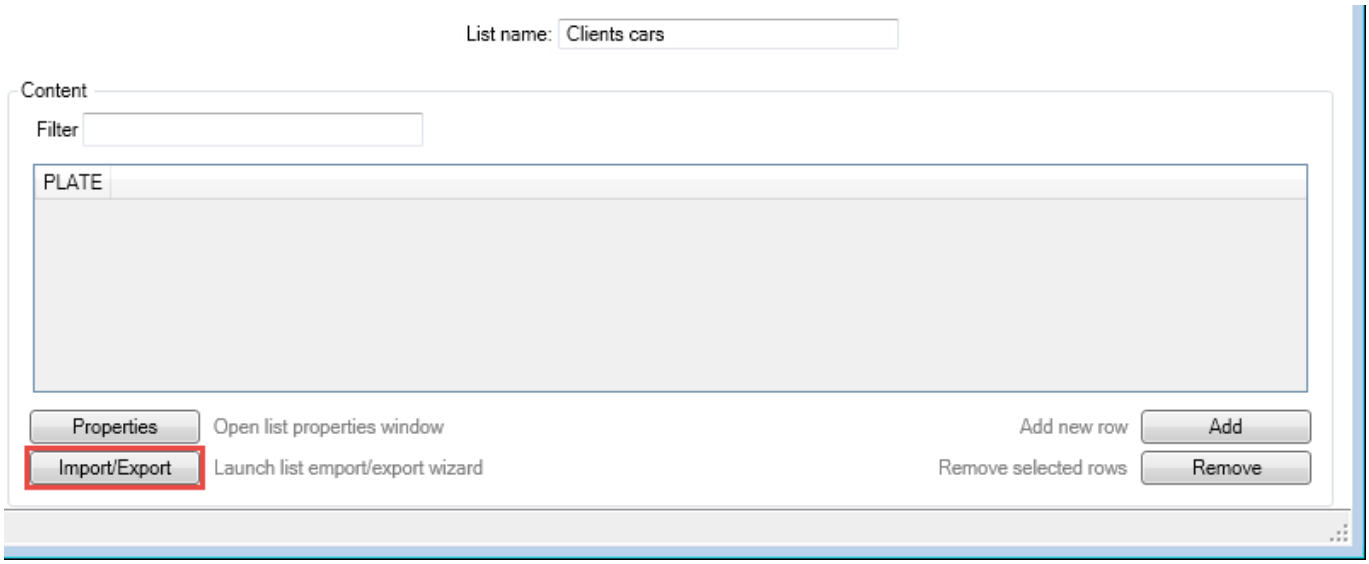
> **Filter** field on the properties panel of the list allows to quickly locate data in the list. The search includes all columns.



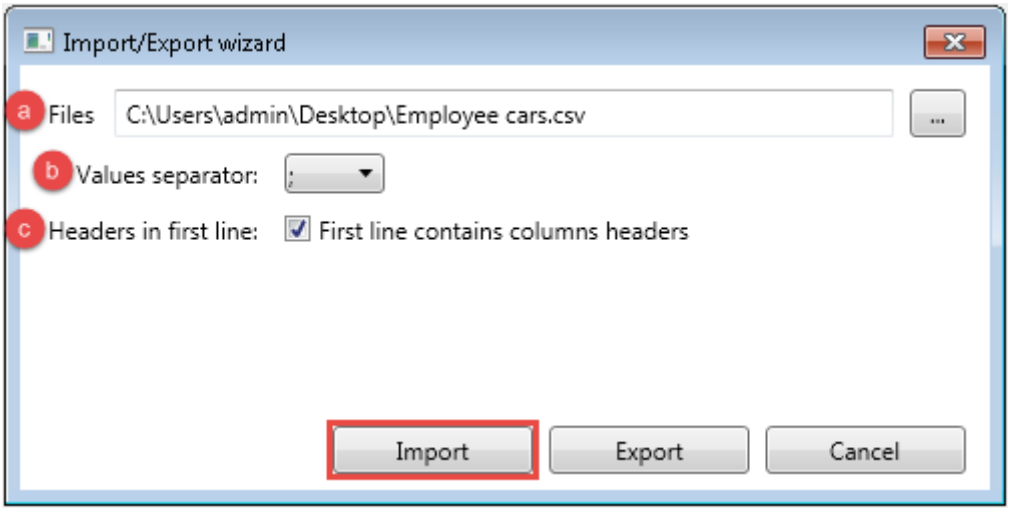
importing the list using a csv file

1. Open properties of the newly created **List** object instance (the "Clients Cars" list would be used further to illustrate the process).
2. Click **Import/Export** button.
3. In the **Import/Export Manager** window that appeared, specify the following:
   - directory of the file to be imported;
   - delimiter-symbol used to denominate button values;
   - whether or not should the import consider the first line as header.

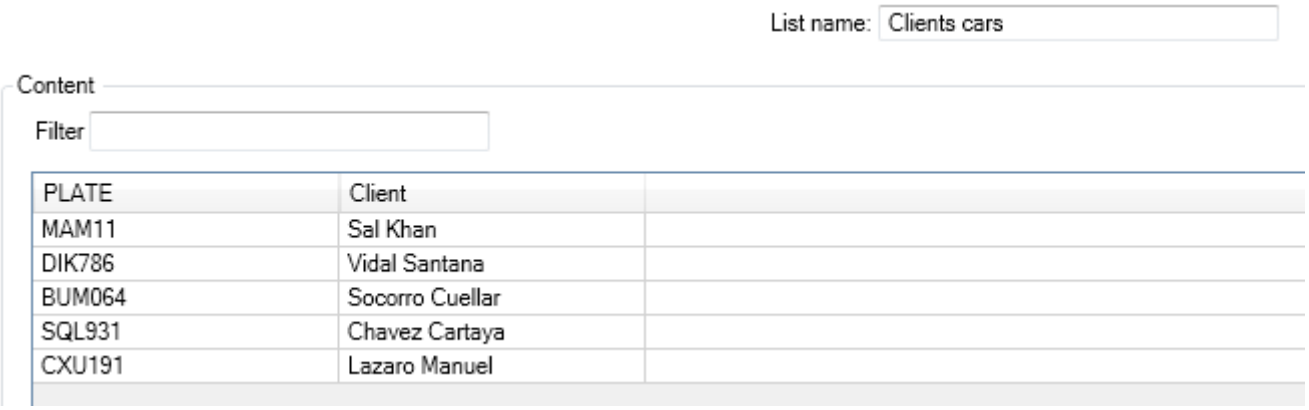
4. Click **Import** button. Save the configuration.



Export to CSV is carried out in the same window. Specify the directory to save the file, name of the file, delimiter (; , or |) and whether or not the header is to be created. After that click **Export** button.



The result is the list being filled with data from a third-party CSV file.



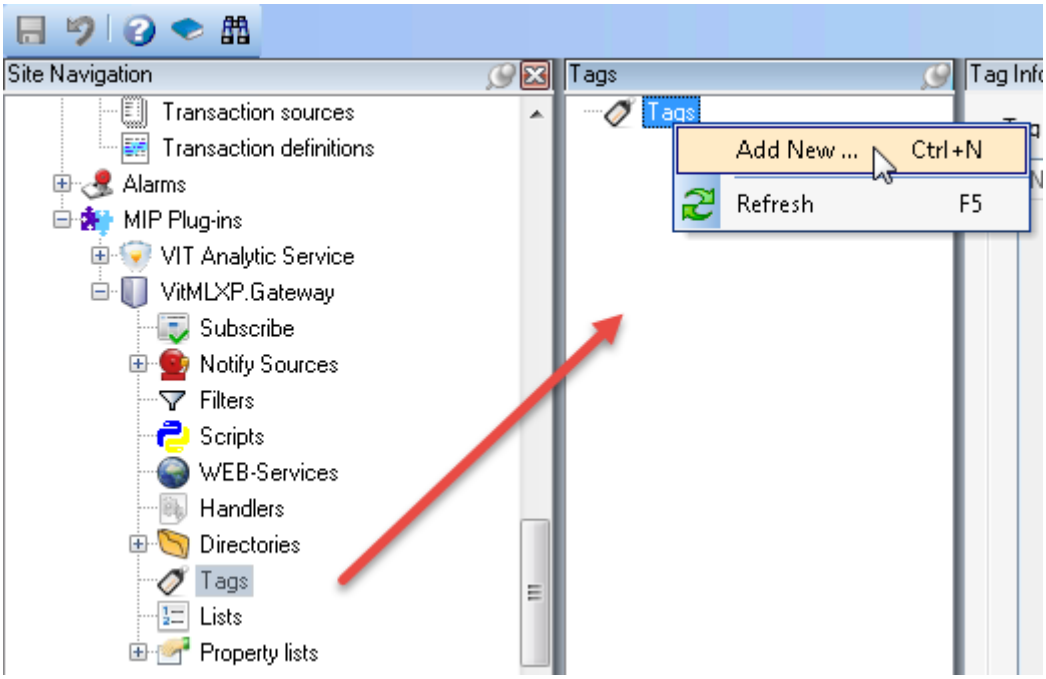
Step 2. Connecting the list to a recognition events source

Tags are used to logically connect objects within the **XProtect** system. A group of objects with the same tag is to be used for the operation of the supplied script.
Logical connection between events source (surveillance camera) and lists means that in the process of license plate recognition, the search would be conducted in all associated lists. If multiple lists are associated with one camera, the search will be conducted in succession until the first match.

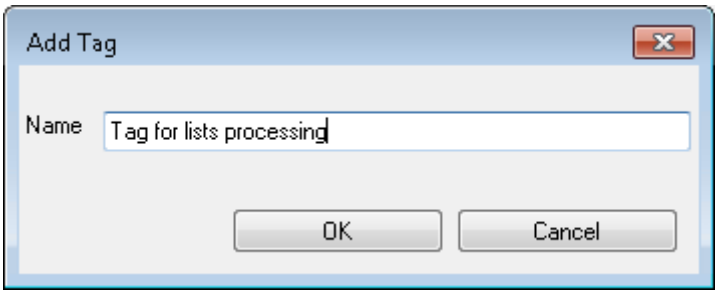
creating tags

1. Select the **Tag** object on the navigation panel.

2. Open the context menu and select **Create New**.



3. Name the new tag in the window that appeared (e.g. "Tag for lists processing").



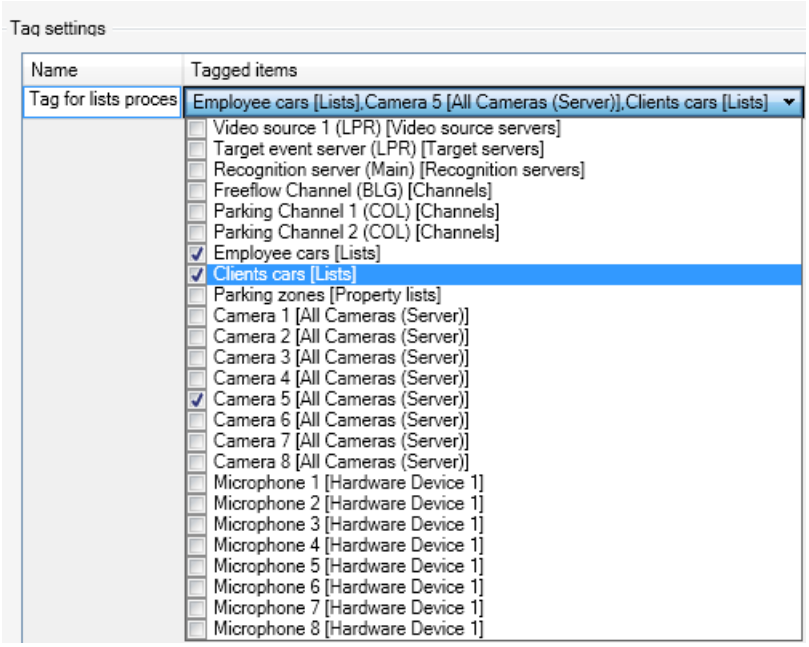
4. Save the configuration.

configuring tags

1. Open properties of the newly created **Tag** object and set the following parameters using dropdowns:

   - the source generating analytic events (e.g. **Camera 1**);
   - one or more license plate lists to be checked during recognition (e.g. "Employee cars", "Clients cars").

2. Save the configuration.



Step 3. Determining special events

The number of analytics events to be determined matches the number of lists to be searched during the recognition. Event name format: [**List name**] **result**.
For example, in previous step, two lists were created: "Employee cars" and "Clients cars". Analytics events for those lists would be named **Employee cars result** and **Client cars result** respectively.

> See the Registering Analytics Events subsection for instruction how to create an analytics event.

Step 4. Importing special scripts

At this stage a script must be imported into **XProtect** system:

- **ScriptForCSV** — used if PLATE is the only mandatory column in the lists to be processed.
- **ScriptForList_Owner** — used if lists contain both **PLATE** and **Owner** mandatory columns.

> An example of script importing is provided in Importing scripts description.

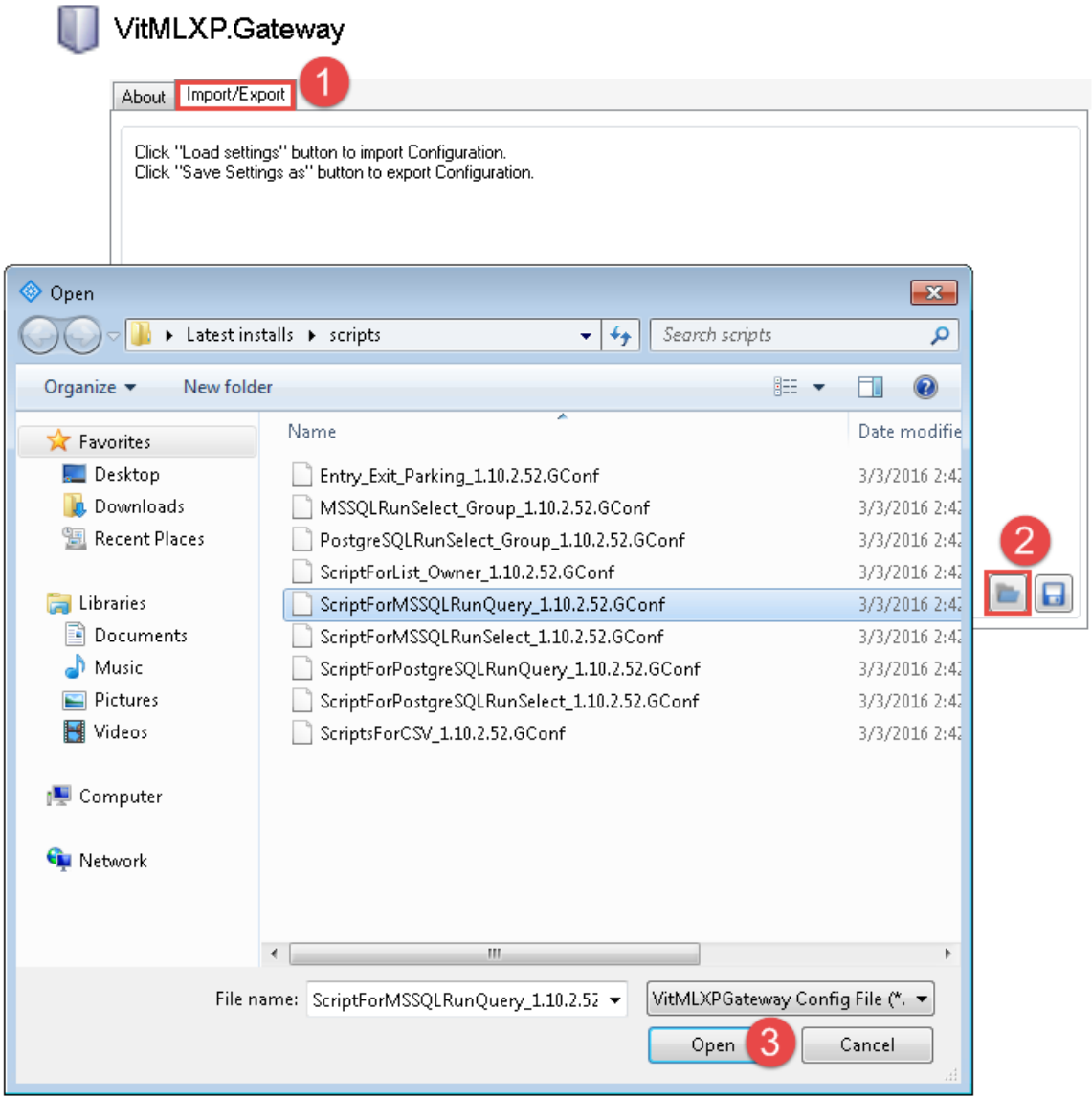
Along with the script, a **FilterForList** filter is imported as well as additional files. The events to be displayed in the client application may be specified on **Parameters** tab of the script editor:

1. [**List name**] **no result** — recognized license plate not found in [list name] list.
2. [**List name**] **result** — recognized license plate found in [list name] list.
3. **Unlisted license plate** — ecognized license plate not found in any of associated lists.

The result of using **ScriptForList_Owner** or **ScriptForCSV** scripts should be aforementioned events being displayed in **XProtect Smart Client**.

External database

Step 1. Importing special script



One of the following scripts is to be imported in order to setup connection to an external database:

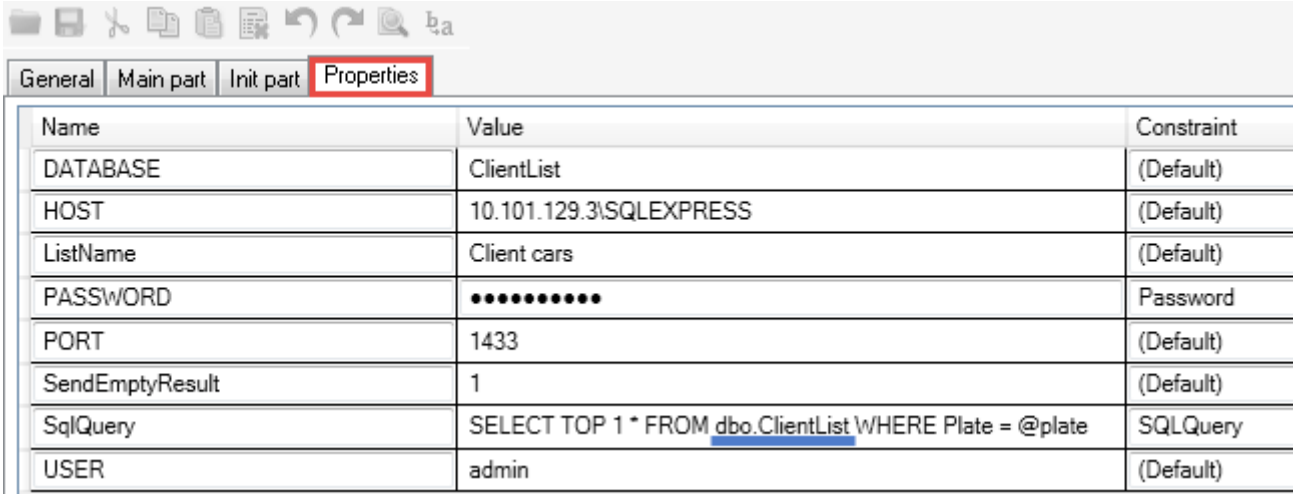
- **ScriptForMSSQLRunQuery** — to connect to a **Microsoft SQL Server** with the ability to manually send requests to the database.
- **ScriptForMSSQLRunSelect** — to connect to **Microsoft SQL Server** using a request previously created by the developer.
- **ScriptForPostgreSQLRunQuery** — to connect to **PostreSQL** with the ability to manually send requests to the database.
- **ScriptPostgreSQLRunSelect** — to connect to **PostreSQL** using a request previously created by the developer.

> An example of script importing is provided in Importing scripts example.

Additionally, **PostgreSQLRunSelect_Group** and **MSSQLRunSelect_Group** scripts may be used to include additional groups into search. For example, while comparing recognized license plates to license plates from "Regular visitors" of the parking lot table, not only the data stored in PLATE column would be included, but also the respective values from **Group** column (employees, clients or suppliers).

Step 2. Configuring script

To configure this script, open its editor window and switch to **Parameters** tab.



The set of parameters for a script used to search for number plates in external databases depends on the ability to create a request manually.
Common parameters (for all scripts):

- **Database** — name of the database used to match the license plate.
- **Host** — address of the host with target external database management system. Record format for PostgreSQL — [IP-address]. Record format for MS SQL — [IP-address] or [IP-address]\[MSSQL server name].
- **ListName** — a sub-string that would act as the first part of name and message of returning events: [**ListName**] **result**, [**ListName**] **no result**. For example, if the search in the database is conducted using a table with license plates from client cars, this parameter may be set to 'Client cars'.
- **Password** — a password to connect to the database management system.
- **User** — the user connecting to the database management system.
- **SendEmptyResult** — parameter to specify whether the event is to be sent if there are no matches with license plate in the database. Possible values: 1 — generate a no-matches event ([**ListName**] **no result**), 0 — do not generate a no-matches event.

Other parameters:

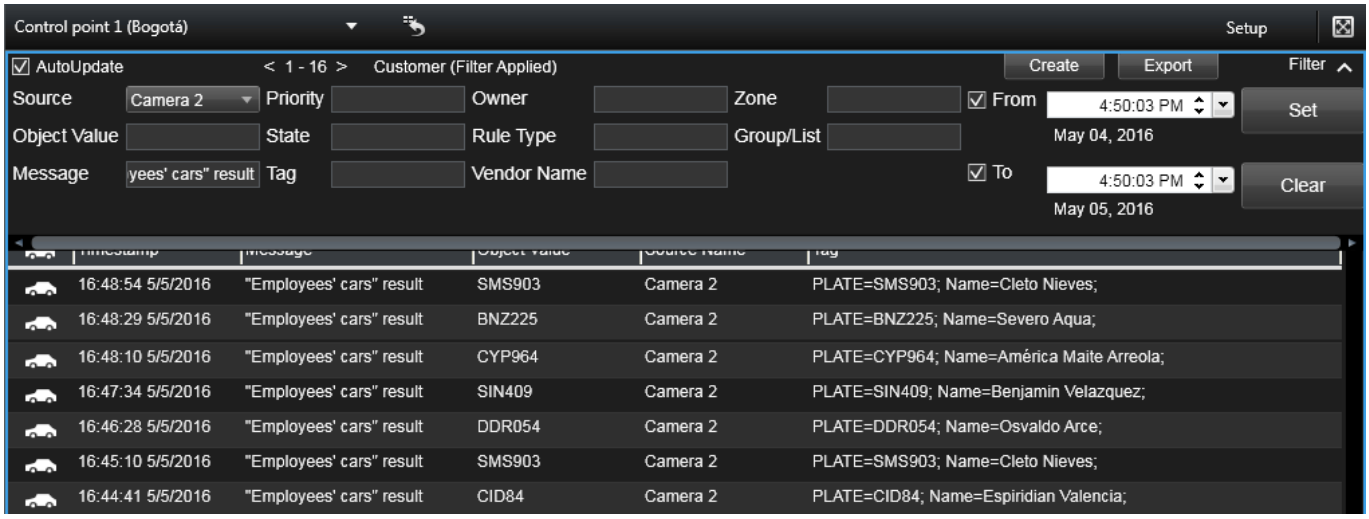
- *using a manually created request:*
  - **SqlQuery** — parameter that uses the request to the database as the value. The primary request (default value of this parameter) may be used by changing the names of target table and column ('Table_Name' and 'Field_Name') in it.
- *using a pre-developed request:*

- **RecordsToReturn** — number of events to be displayed if duplicates found in the database. For example, there are 4 duplicate records in the database. If this parameter is set to 1, client application would only display 1 event of the license plate being found in external database.
- **FieldName** — name of the column to store license plates in the database.
- **TableName** — name of the table to store license plates in the database.
- **GroupName** — name of the database column that contains additional values to group the data (this field is active if either **PostgreSQLRunSelect_Group** or **MSSQLRunSelect_Group** is used).

> To have [**ListName**] **result** and [**ListName**] **no result** events displayed in **Alarm manager**, an alarm event must be configured in **Management Client**. Trigger event parameter must be set to [**ListName**] **result** if the license plate was found in the database or [**ListName**] **no result** if the license plate was not found in the database.

Processing the results

Using the script should result in search results (if recognized) being displayed in **XProtect Smart Client**.



A TROUBLESHOOTING

introduction

If after being configured the LPR system does not function properly, refer to checklists provided in this section as they may indicate an error in channel settings.

- The **Source** field on **LPR** tab in **VIT Analytic Server Manager** has 'Connecting' value while either **Source FPS** or **Recognition FPS** is set to 0 (Fig. 28).
- The target server field on **LPR** tab contains 'Error' value (Fig. 29).
- **LPR** tab contains no information about recognition channels.
- No events and associated alarms sent after the display of recognition results in **XProtect Smart Client** was configured.

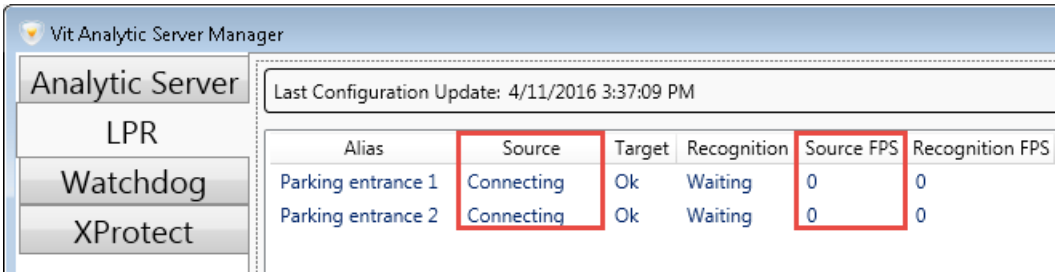


Figure 28: Error indication: Example 1

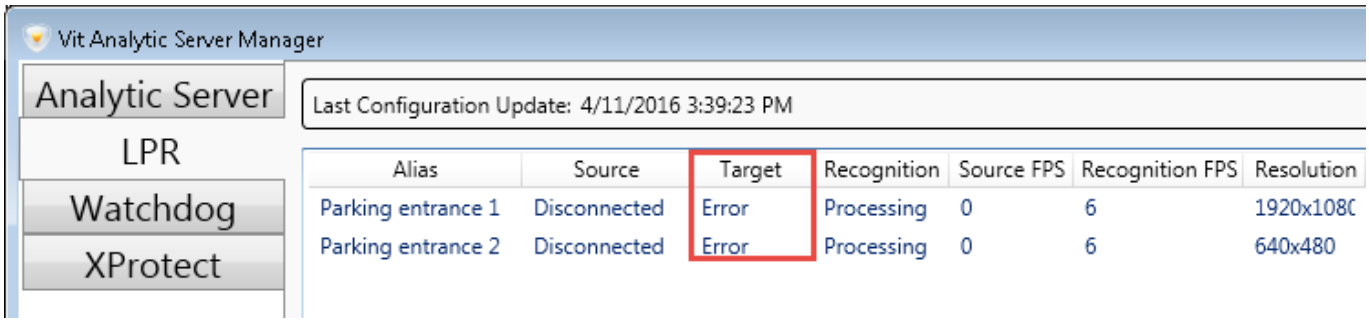
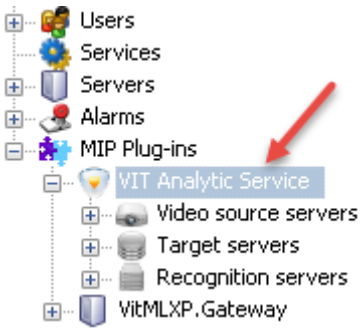


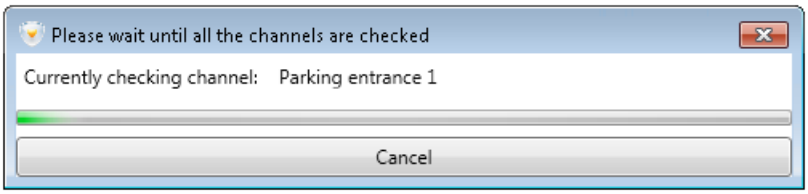
Figure 29: Error indication: Example 2

> This section does not provide any recommendations on improving the quality of videostream and license plate images. Such recommendations may be found in 'License Plate Recognition Cameras Selection, Setup and Installation Guide' document.
> The checklists provided in this section are useful for detecting configuration errors and licensing problems.

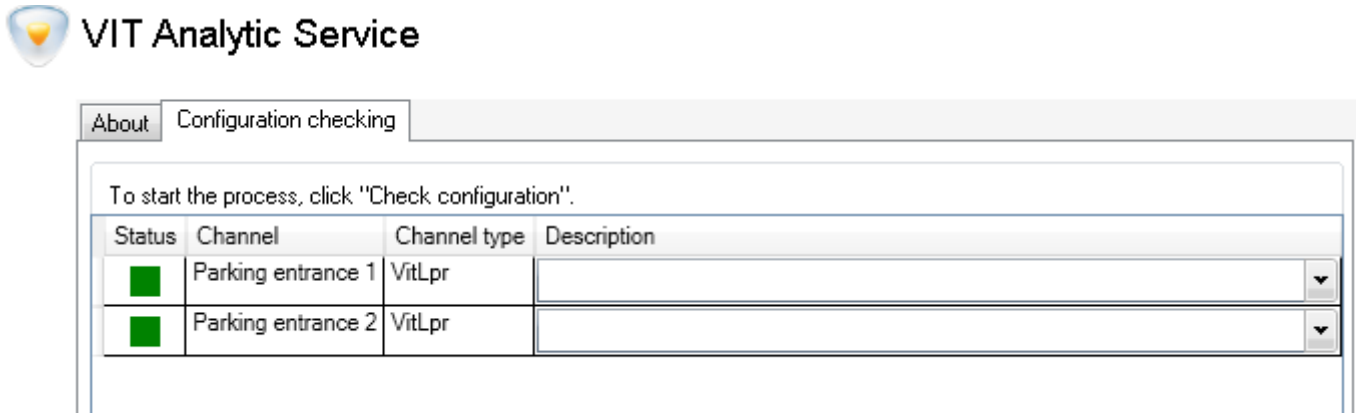
A special tool located in **VIT Analytic Service** parent object may be used for troubleshooting.



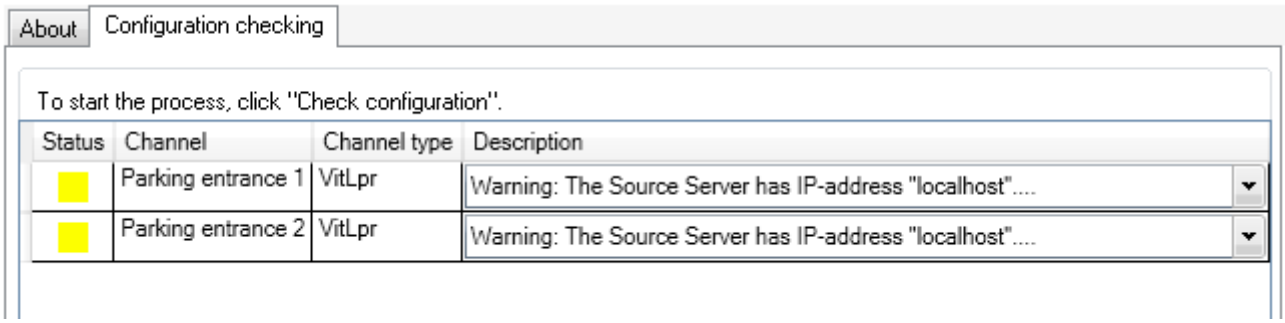
When the object is opened, switch to **Configuration checking** tab. If the tool is opened for the first time (or after the **Management Application** was rebooted) it will not show any data. Click the **Check configuration** button to analyze the settings of all channels in the system and display the possible configuration errors.



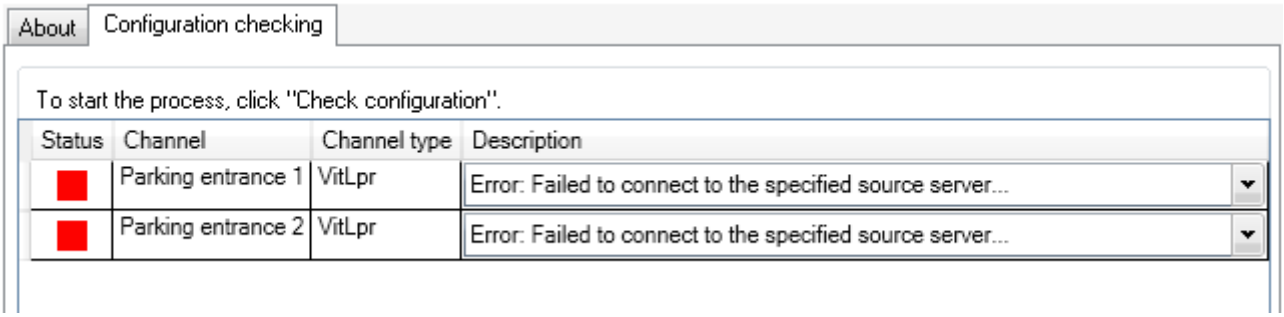
If the settings are correct, the **Status** field of a specific channel line would turn green and the **Description** field would contain no records of errors.



If present settings may cause issues during recognition or the results may not be sent to the event server, the **Status** field of a specific channel line would turn yellow and the **Description** field would contain the alert text.



If present settings make recognition impossible, the **Status** field of the channel would turn red. The **Description** field would contain description of what caused the issue as well as troubleshooting recommendations.



Additionally, it is recommended to perform a manual check using the checklists provided further.

recognition sub-system settings: checklist

- **AutoCode XProtect** modules have been installed in correct order ☐

  > If **Analytic Server** and **LPR** modules were installed in incorrect order, the errors of connection to **XProtect** may not be displayed. Additionally, both **LPR** and **Watchdog** tabs in **Analytic Server Manager** would not display information about connected channels. The **LPR** tab may not be displayed as well.

- The video source servers are enabled. The correct host IP-address, port, password and login are inserted. During the configuration, video source server connection was successful (the 'Connected' value is shown after the **Test** button is clicked) ☐
- The target servers are enabled. Correct host IP-address and port for receiving events are inserted ☐
- Correct IP-addresses are inserted into parameters that require host IP-addresses. If some parameters are set to default IP-addresses such as 'localhost' or '127.0.0.1', the correct addresses are to be inserted ☐
- If **XProtect** event server and **Analytic Server** are located on different hosts, the **Configurator** module must be installed on a host that contains **XProtect** event server (without the mandatory installation of **Management Client**) ☐
- **XProtect** event server is enabled ☐
- The usage of analytics events in **XProtect** is activated. **Plate detected**, **Car detected**, **Car lost** events are created ☐
- The specified recognition server is activated ☐
- The specified channels are active ☐
- Correct sources of video stream (video source server, camera) and target servers are specified in channels configuration ☐
- Each channel has license plate templates selected that correspond to the type and issuer-state of license plates found in frames to be processed ☐
- The size of the frame for maximum possible image of a license plate in frame (specified when configuring the channel of **Recognition settings** window) is set to 110-200% of the average actual size of the license plate ☐

- Correct directions are set for vehicles in frame using the direction filter when configuring the channel in **Recognition settings** window ☐
- Recognition zones are specified when configuring the channel in **Recognition settings** window ☐

> Often, the incorrectly specified direction can be located quickly if the host has multiple recognition channels. In such cases, the issues occur on one or more specific channels. Direction filter settings may not correspond to the actual direction of vehicles in frames received from a new camera if, for example, the equipment used on the channel was replaced.

license: checklist

- The license key is connected to a computer if hardware protection is used ☐
- Recognition server has **Sentinel LDK** (**Sentinel Runtime**) license key driver installed ☐
- Service software (Windows Firewall, anti-virus software) does not block the license key (**hasplms.exe**) and Port 1947 ☐

  > C:\WINDOWS\system32\hasplms.exe and Port 1947 may be added to the list of exceptions blocking.

- The feature-set specified in the license matches the list of features required ☐
- The license key is not installed on a computer with terminal software ☐
- License has not expired ☐
- System time was not changed on recognition server (if a software key is used) ☐

B MESSAGES

events

Event is a message generated after specific actions in the system. In **XProtect** system, the events are mainly classified by their source. In its relation to the system, an event may be either internal or external.
Events are used for:

- exchanging information between **XProtect** components and third-party applications/devices.
- creating system reactions to be triggered as a response to events.

Event Server is a service within **XProtect** used for:

- recording events to the database.
- providing access to events.

XProtect Event Server allows MIP-Plug-ins[1] (including **AutoCode XProtect** recognition sub-system) and third-party applications to communicate with **XProtect** system. The following event types are used:

1. **Analytics events**
   Messages generated by MIP-Plug-ins such as external third-party video content analysis (VCA) providers. They are sent to **XProtect** system as a result of plug-ins operation (e.g. along with video processing data in real time). It is possible to:

   - enable/disable receiving analytics events in **XProtect**;
   - specify the port for receiving analytics events;
   - allow receiving events from all network addresses or specify trusted addresses[2].

   > Instructions for setting up analytics events:
   >
   > - **XProtect** administrator's manual:[a]
   > - Add and edit an analytics event
   > - Test analytics event
   > - Control events reception using **Options** window (**Options** dialog box, **Analytics Events** tab)
   > - This administrator's manual (see Registering Analytics Events):
   > - Determining analytics events (**Plate detected**, **Car detected**, **Car lost**) generated by **LPR** module when processing the video as well as their dependence on selected recognition settings
   > - Registering the events

[1] Third-party software integrated into **XProtect** to expand the feature-set of the system. The plug-ins are created using **Milestone Integration Platform** (**MIP SDK**) software development tool.
[2] **Options** window of **Management Client**

[a] Hereinafter, the manuals for the versions of the system that support third-party MIP-products integration: **Express**, **Professional**, **Enterprise**, **Expert**, **Corporate**

2. **Generic events**

   Represent the conditions that the strings received via TCP- and UDP-ports (from sources that are third-party in relation to **XProtect**) must satisfy. Generic events enable messages to be received from external devices/applications (e.g. access control systems, emergency alarms).

   The system analyzes incoming strings on the subject of satisfying the condition created when the generic event was created. If the condition is met, the system generates a reaction to the event. A respective event is the source of the generated reaction.

   Instructions for setting up generic events:

   - **XProtect** adminisrator's manual:
     - Adding a generic event and configuring it (see 'Add a generic event' and 'Generic event (properties)')
     - see "Test a generic event"
   - This administrator's manual (see Receiving Events from Third-Party Sources):
     - Setting up the reception of generic events from third-party sources[a] via **Analytic Server** module to be used in **AutoCode XProtect** and/or later redirected to **XProtect** system.

   [a] as well as sending analytics events generated based on received generic events to **XProtect** system. This occurs for tasks that require more precise time measurements (less than seconds) when generic events are generated by **XProtect** system.

Reactions to events (including alarms) are set with the help of **rules**. A rule is a set of system actions to be performed under specific conditions (e.g. start video recording on Camera 1 if it registered movement in frame).

alarms

Alarms management functionality is a component of **XProtect** event server. Both alarms and events are stored in the same database.
Alarms can be generated based on the following:

- system (internal) events such as 'Not enough disk space', 'Movement detected' or 'Server not responding';
- external events such as analytic events from **LPR** and **Gateway** modules.

On the client side, the alarms are processed and delegated in **XProtect Smart Client**.

Instructions for configuring alarms:

- **XProtect** adminisrator's manual:
  - Adding and configuring rules (see 'Rules')
  - Enabling alarms processing based on user roles (see 'About alarm configuration')
  - Adding and configuring a new alarm (see 'Alarm Definitions')
  - Configuring metadata (e.g. priority level) and processing properties of the alarm (see 'Alarm Data Settings')

> – Configuring sound notifications (see 'Sound Settings')

list of autocode xprotect analytics events

1. **Plate detected**

   The fact of receiving the best possible recognition result. Source: **LPR** module.

   Depends on **Send event while car in frame** setting of the **Channel** object:

   - If this setting is *deactivated*, the event is generated after the vehicle has left the frame (along with the **Car lost** event).
   - If the setting is *activated*, the event is generated when the vehicle enters the frame (along with the **Car detected** event).

2. **Car detected**

   The fact of receiving the first recognition result registering a vehicle entered the frame. Source: **LPR** module.

   It depends on **Result Delay (msec)** setting of the **Channel** object. The event is generated after the time specified in the setting.

3. **Car lost**

   The fact of the license plate leaving the recognition zone. Source: **LPR** module.

   Depends on **Check license plate overlap (sec)** setting. If the license plate does not reappear in the frame within the time period specified in the setting, it is considered to not be overlapped, so the vehicle has left the frame.

4. **Plate corrected**

   Recognized license plate was manually edited in **XProtect Smart Client**. Source: **Viewer** module.

5. **Plate added**

   An event with the following information was created manually in **XProtect Smart Client**:

   - license plate;
   - event source.

   Source: **Viewer** module.

6. **PASS**

   Registering a vehicle entering/exiting the parking zone. Source: **Gateway** module.

   Generated by **Entry\Exit_ParkingPASS** script.

7. **DOWNTIME**

   Calculating the time vehicle spent on the parking. Event contains the name of the parking zone, entrance/exit time and time spent in it. Source: **Gateway** module.

   Generated by **Entry\Exit_ParkingDOWNTIME** script.

8. **vitml.error.acs.pass_flap**

   Error message. The time spent on parking is *less than* the specified minimum time (a duplicate recognition event is possible). Source: **Gateway** module.

   May be generated by either **Entry\Exit_ParkingPASS** and **Entry\Exit_ParkingDOWNTIME** scripts.

9. **vitml.error.acs.port_incorrect**

   Error message. Registering a vehicle entering the zone that was not specified in property list settings. Source: **Gateway** module.

   May be generated by either **Entry\Exit_ParkingPASS** and **Entry\Exit_ParkingDOWNTIME** scripts.

10. **vitml.error.acs.lost_inpass**

    Error message. A vehicle with the license plate that was not registered when entering the parking zone is exiting the parking. Source: **Gateway** module.

    May be generated by either **Entry\Exit_ParkingPASS** and **Entry\Exit_ParkingDOWNTIME** scripts.

11. **vitml.error.acs.lost_outpass**

    Error message. The maximum allowed time on parking is exceeded (entrance registered, exit not registered). Source: **Gateway** module.

    May be generated by either **Entry\Exit_ParkingPASS** and **Entry\Exit_ParkingDOWNTIME** scripts.

12. **[List name] result**

    Recognized license plate found in [list name] list. Source: **Gateway** module.

    Generated by either **ScriptForList_Owner** or **ScriptForCSV** scripts.

13. **[List name] no result**

    Recognized license plate not found in [list name] list. Source: **Gateway** module.

    Generated by either **ScriptForList_Owner** or **ScriptForCSV** scripts.

14. **Unlisted license plate**

    Recognized license plate not found in all associated lists. Source: **Gateway** module.

    Generated by either **ScriptForList_Owner** or **ScriptForCSV** scripts.