A MITEL PRODUCT GUIDE

Mitel SIP-DECT 10.1 Event Manager

System Manual

Version 1.0

Mitel

NOTICE

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Networks™ Corporation (Mitel®). The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes.

TRADEMARKS

Mitel SIP-DECT 10.1 Event Manager

System Manual

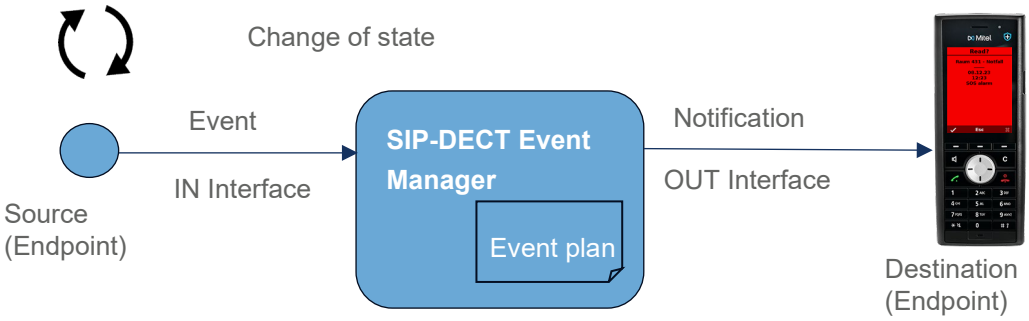
Release 10.1 – December 25

Table of Contents

Overview

Introduction

The SIP-DECT Event Manager is an integrated software component of a Mitel SIP-DECT system. It is used for the automated processing of incoming events and the sending of outgoing notifications. The SIP-DECT Event Manager can process events from various sources, including SIP-DECT terminals, the SIP-DECT system itself, and other external systems. The processing of the events is carried out according to user-defined rules set by the administrator.



The primary flow is to send notifications as text messages to SIP-DECT phones, which are triggered by incoming events. In this way, SIP-DECT supports customer workflows beyond voice calls, e.g., text messages can be sent to DECT phones to inform about events from nurse call systems without the need for additional hardware.

Processing rules for different types of events consist of event plans, their event phases, notification profiles and different types of confirmation requests.

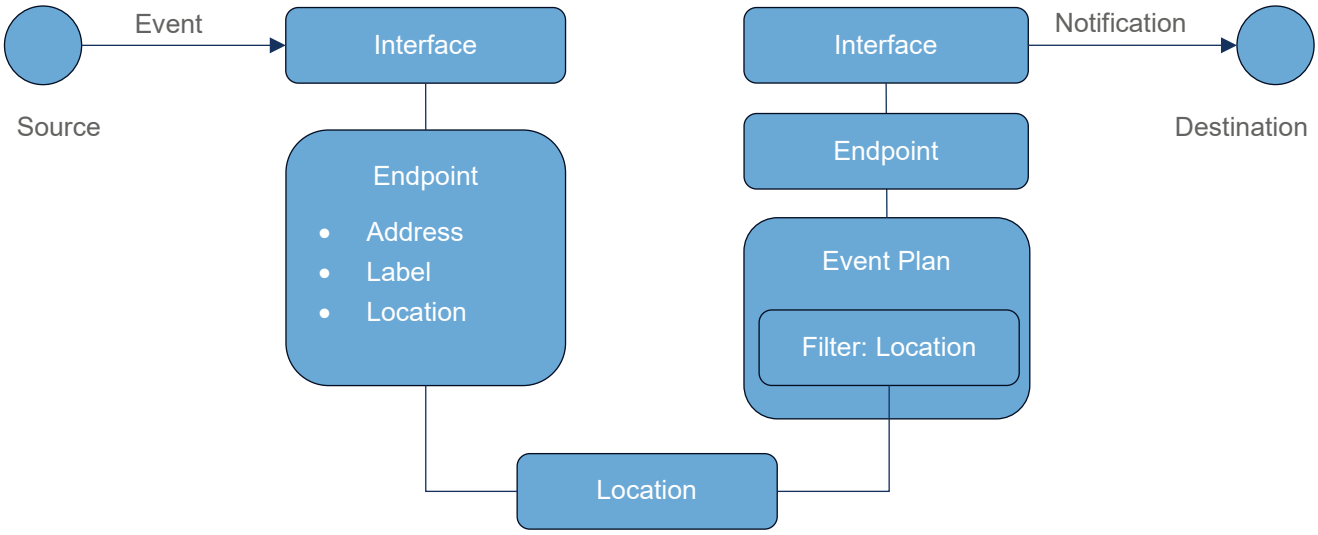
If there is a change in status, e.g., a key press, a source sends an event to the SIP-DECT Event Manager via an input interface. The SIP-DECT Event Manager generates notifications, e.g., text messages, and sends them to destinations, e.g., DECT telephones via outgoing interfaces according to a suitable event plan.

Some interface types are only incoming or only outgoing interfaces, and some can be both incoming and outgoing.

Sources and destinations are called endpoints. They are assigned to the interfaces through which they communicate with the SIP-DECT Event Manager. Endpoints have a unique identification e.g. a telephone number.

Endpoints are also assigned to locations. Depending on the location, a specific event plan can be selected. This allows the same event to be treated differently depending on where it originated.

The following illustration is intended to visualize the relationships between endpoint location and the event plan location filter.



The Event Managers DECT and BLE Locating supplements the Event Manager functionality described above with a textual and graphical display of the position of a DECT device based on:

- the DECT radio coverage by a base or
- Bluetooth Low Energy (BLE) beacon signal

A graphical display is provided in a detailed and an overview view in the event of an emergency call, triggered by pressing the SOS button on the Mitel DECT telephone (722dt, 732d, 742d, 632d(t) V2) or by a sensor alarm of the DECT device (732d, 742d, 632d(t) V2) as well as feature access codes for customer-specific configurable alarm triggers. In addition, the position of a locatable DECT device can also be queried independently of an event.

The DECT radio coverage by a base station is typically approx. 30 to 50 meters in buildings depending on the structural conditions and approx. 300 meters in free field.
To increase the accuracy of the locating information, Bluetooth-enabled Mitel 700d DECT phones (722dt, 732d, and 742d) can use the signal from Bluetooth Low Energy (BLE) beacons. These must support the iBeacon protocol. Due to the lower and adjustable transmission power of BLE beacons, the radio coverage can be selected to meet the requirements of more accurate locating.

For the graphical display of the position of a DECT device in case of an event, a suitable event plan must be configured and consequently the triggering DECT phone must be set up as an endpoint.
A locating button is offered in the monitor in the locating section, which opens the graphical display.

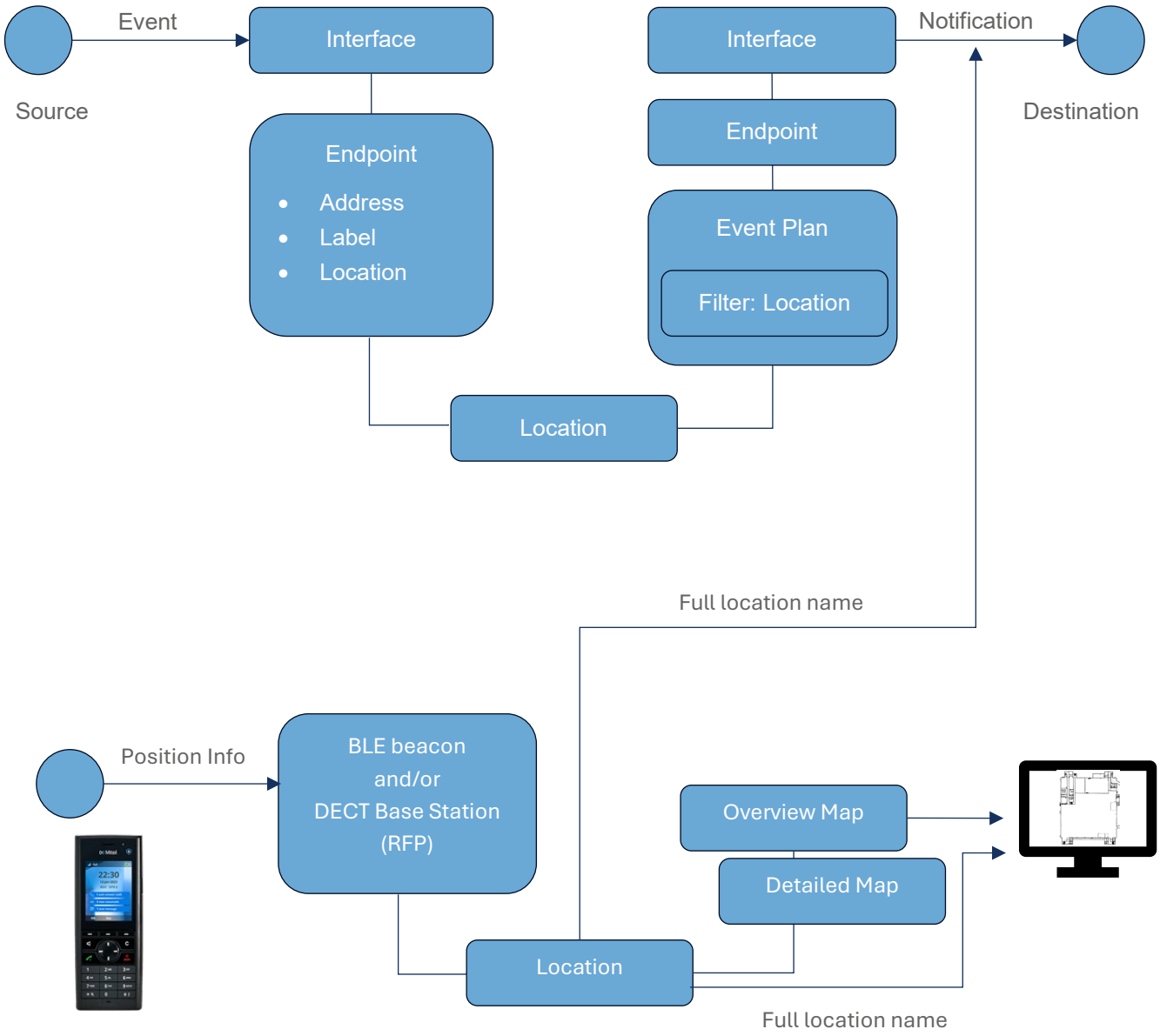
Attention: The appropriate event plan is selected based on the configured location and not on the position determined via DECT or BLE.

No event plan is required for an event-independent determination and display of the position of the DECT phone of a locatable and possibly trackable user. The telephone user list in the Locating section, which contains all locatable users, can be used for this purpose. A locating button is also provided here.

To determine the location of a DECT telephone, the DECT base stations and BLE beacons must be assigned to locations. A map must also be assigned to the location and the location must be positioned on a detailed map and on an overview map for graphical representation.

If SIP-DECT locating is used, the full location name from the Event Manager is used in the notification instead of the base station data Site, Building, Corridor etc. configured in the OMM. This ensures that the textual location in notifications matches those in the Event Manager web GUI.

The following figure illustrates the relationships between base stations, maps and locations, as well as the event handling in the Event Manager.



A Linux Server installation of the Event Manager is required to support DECT and/or BLE locating.

Attention: It is recommended not to plan and set up locations too granularly, as DECT works with larger and overlapping radio fields.

Where is the SIP-DECT Event Manager running?

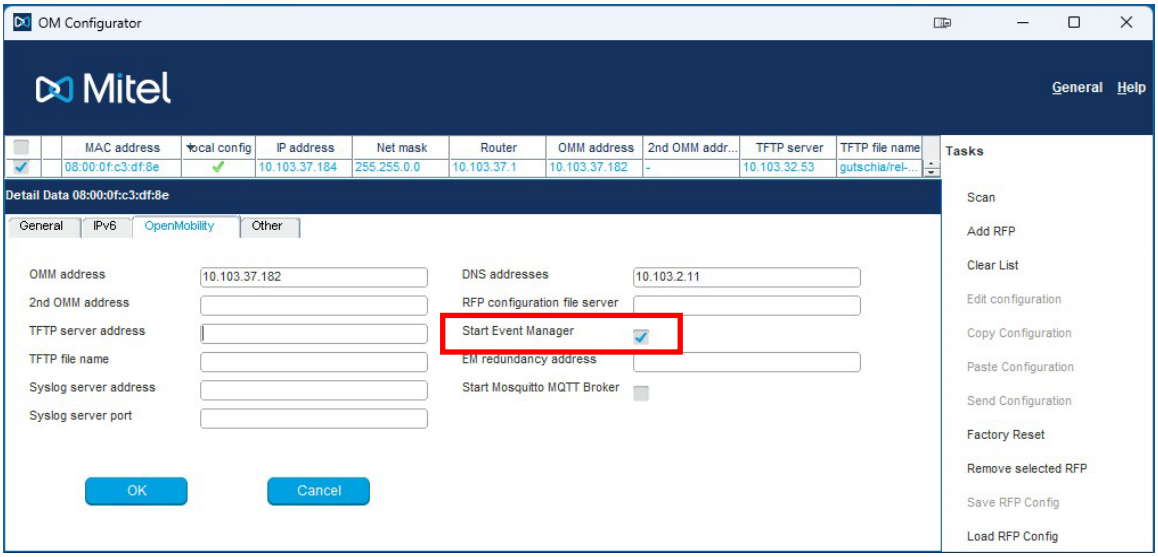
4th generation RFP

The SIP-DECT Event Manager can run on a 4th generation RFP (RFP44, RFP45, RFP47 or RFP48 WLAN) and is part of the iprfp4G.dnld SW package.

The SIP-DECT administrator determines in the OMC (OM Configurator) on which RFP the SIP-DECT Event Manager is started. This allows a different RFP to be used than the RFP used by the OMM, so that the OMM and SIP-DECT Event Manager do not compete for the same resources.

This also implies that the SIP-DECT Event Manager RFP (the RFP on which the SIP-DECT Event Manager runs) has a local static IP configuration. This ensures that the SIP-DECT Event Manager can be started independently of other services and is always accessible under the same IP address, as is usual for services. Only one SIP-DECT Event Manager per SIP-DECT installation is supported.

To start the SIP-DECT Event Manager the “Start Event Manager” flag must be set as shown below.



If this “Start Event Manager” flag is removed again from an RFP via the OMC, the Event Manager will be stopped, and its database will be removed only during the next start of this RFP.

> Please note: The Event Manager on an RFP can only handle configurations within the configuration limits of an RFP OMM, i.e. max. 256 RFPs and max. 1024 DECT users. If the OMM runs on a Linux server, the Event Manager must also run on a Linux server.

Linux Server

The Event Manager can also be installed as an application on a Rocky Linux® 9. A rpm file is available for this purpose. The rpm file is also part of the SIP-DECT VM images. After the initial start of a VM, the OMM, MOM or the Event Manager can be installed. Information on this can be found in the SIP-DECT LINUX Server Installation administration guide.

The Linux Server installation of the Event Manager supports DECT and BLE locating with a textual and graphical representation of the position of a DECT device. Please see section DECT and BLE Locating! Otherwise, the EM on a Linux server does not differ from an EM on an RFP with regards to the scope of

features.

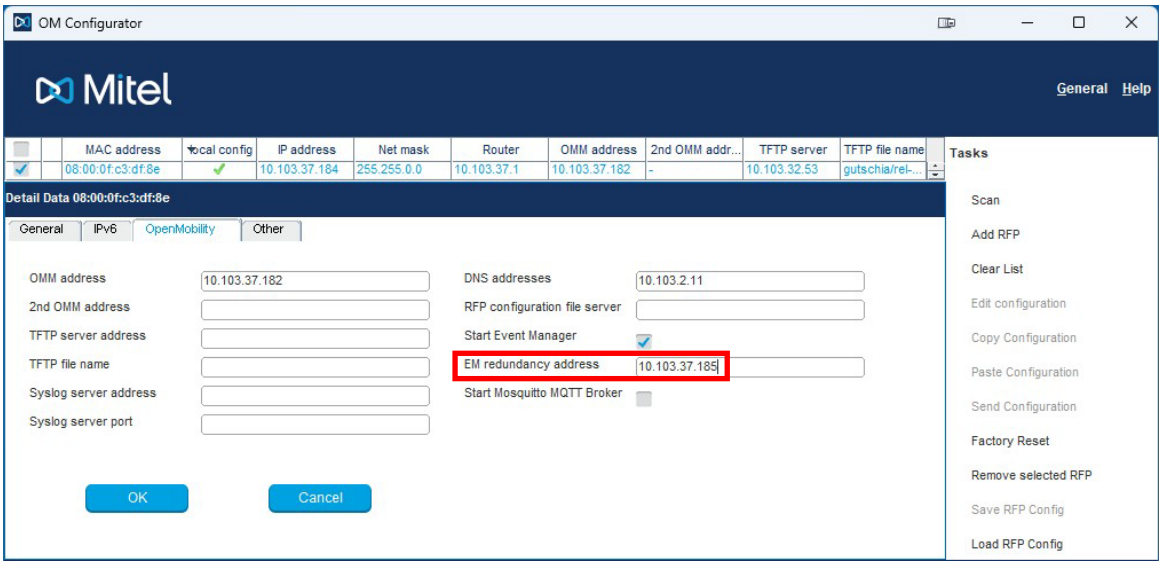
Since the Mitel CloudLink daemon is not available for server installations of the Event Manager, the remote management is not available in this case.

SIP-DECT Event Manager redundancy

Up from release 10.1 there can be configured a redundancy function of the Event Manager to ensure greater availability. This functionality is available on 4G-RFP as well as on server installation and is based on timestamps of the databases of both Event Manager instances. During the installation of the Event Managers on RFPs these RFPs must be configured in the OMM and connected to it to ensure that both instances uses the same time base and the timestamps are comparable. In case of a running Event Manager instance with a former release version there must be made the software update before configuring the redundancy feature. The Event Manager must be restarted once with the new software release to ensure that the correct timestamp is written into the database. After that the configuration of redundancy feature can be done.

Redundancy on 4th generation RFP

The redundancy feature on 4th generation RFP can be configured in the OMC (OM Configurator) by setting of an EM redundancy address.



Redundancy on Linux server

The redundancy feature on a Linux server can be configured in '/etc/sysconfig/SIP-DECT-EM' configuration file which must be adapted in the following way:

Remove the # in one of the two lines with IPv4 or IPv6 addresses and replace the IP addresses with the addresses of the both server instances!

if you use redundancy for EM activate parameter below with EMs IP addresses
#EM_REDUNDANCY="192.168.0.1+192.168.0.2"
#EM_REDUNDANCY="fdc0:a8::1+fdc0:a8::2"

Redundancy feature in praxis

If redundancy feature is configured, the following happens after the start of an Event Manager instance:

- Up to 30 seconds the EM instances will try to connect via TCP port 16333 to the configured

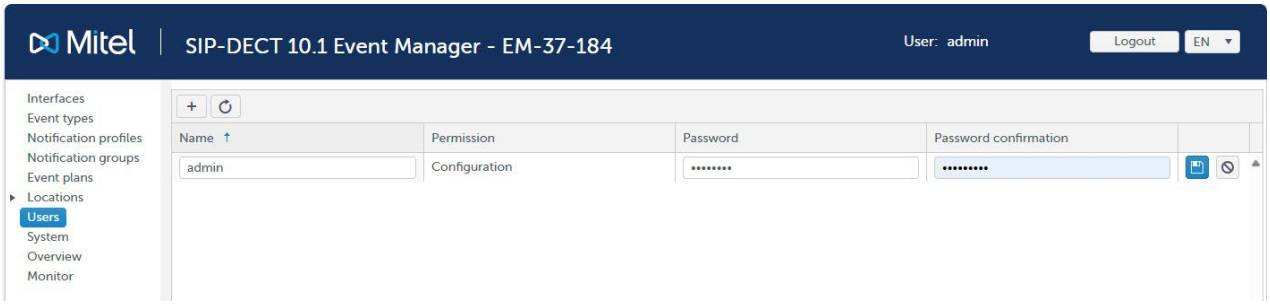
redundancy instance. If there is no connection established within this period the EM starts as active Event Manager and activates the saved configuration

- Also after this 30 seconds period the active Event Manager will continue trying to connect to the configured redundancy instance.
- When connection could be established, the two instances will exchange their redundancy status:
    - Status of EM instance (active / passive)
    - Timestamp of own database
    - Target type (RFP, Server)
    - Software version
    - Uptime (active instance: how long running)
    - Configured IP addresses (own / remote)
- Based on the exchanged data the EM instances decide who should be the active and passive instance and communicate this decision to the other instance
- If both instances got the same result, they will start as active and passive instance. The following cases are available:
    - Both EM instances are active
      If the timestamps of both databases are different, the instance with the newer database keeps active, the other one restarts
    - One EM instance is active
      Depending on the timestamps of the both databases will be decided, which instance shall be the active one and the status keeps as it is or both instances restart to continue as active and inactive. If both timestamps are equal, the instances keep running as before.
    - Both EM instances are passive (e.g. in the first 30 seconds after the start)
      Depending on the timestamps of the databases will be decided, which instance shall be the active one and both instances start as active or passive. If both timestamps in the databases are equal, the instance with the lower IP address will start as the active one, and the other instance stays passive.
- If all cases are clarified, the active instance will transfer the actual database to the passive instance to synchronize the data.
- An interruption of the TCP connection on port 16333 between the two instances will lead immediately to activation of the passive instance. If the connection can be reestablished, the process will start again as described before.
- Changes in the database of the active instance will be transferred to the passive instance (with actual timestamp) and saved there also.
- During the import of a saved database file into the active instance the timestamp of the database will be patched with the current time and then transferred to the passive instance.
- During a restart of the active Event Manager instance with factory reset, the databases of both instances will be deleted. The active instance restarts and the passive instance will be active immediately with an empty (default) database. After the completion of the restart, the scenario of detecting the active and passive instance will be started new.
- The connection of the two instances will be monitored by heartbeat messages every 5 seconds.

Accessing the SIP-DECT Event Manager

The SIP-DECT Event Manager has its own web administration interface which is available via https on port 8444 - https://<IP address>:8444.

Use **admin** as the username and password to login for the first time. During login for the first time, the user is asked to change the password.



License Requirements for the SIP-DECT Event Manager

The SIP-DECT Event Manager requires a license for the configured and activated endpoints. There is a built-in license available already for 5 endpoints.

For additional endpoint licenses a SIP-DECT license is required which covers the amount of configured SIP-DECT Event Manager endpoints. It is strongly recommended to import this license into the OMM before the configuration of the Event Manager.

If the number of configured SIP-DECT Event Manager endpoints exceed the number of licensed endpoints, a warning is displayed on the administrator web interface and notifications are sent to various randomly selected SIP-DECT endpoints every 15 minutes. These notification messages are not monitored by the Event Manager and could not be deleted from within the application (also in case the license would be updated to cover the configured number of endpoints). The notifications will be visible on the SIP-DECT terminals as long they are not read and deleted on the terminals itself.

The SIP-DECT Event Manager uses advanced SIP-DECT messaging and alerting features without requiring a “Mitel SIP-DECT Messaging & Alerting License Enterprise” license.

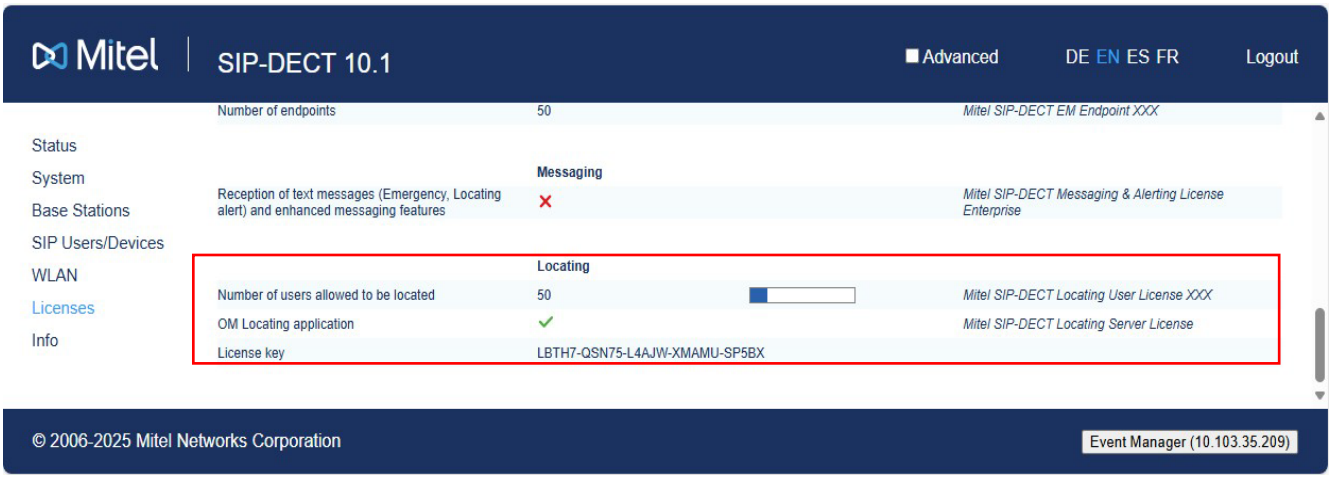
The SIP-DECT Event Manager provides location information for SIP-DECT alarm trigger e.g. SOS-Key or Man-Down automatically without requiring locating license “Mitel SIP-DECT Locating User License XXX”. For this purpose, the Event Manager uses the site, building, corridor etc. information of the base station configured in the OMM.

License Requirements for the DECT and BLE locating functionality

To use the locating functionality the following SIP-DECT licenses are required

- Mitel SIP-DECT Locating User License XXX or Mitel SIP-DECT BLE Locating User License XXX
- Mitel SIP-DECT Locating Server License

Systems with Locating Licenses from older releases will only support DECT locating, no BLE locating



In case of a new or upgraded license for release 10.1 it may be includes BLE locating functionality.

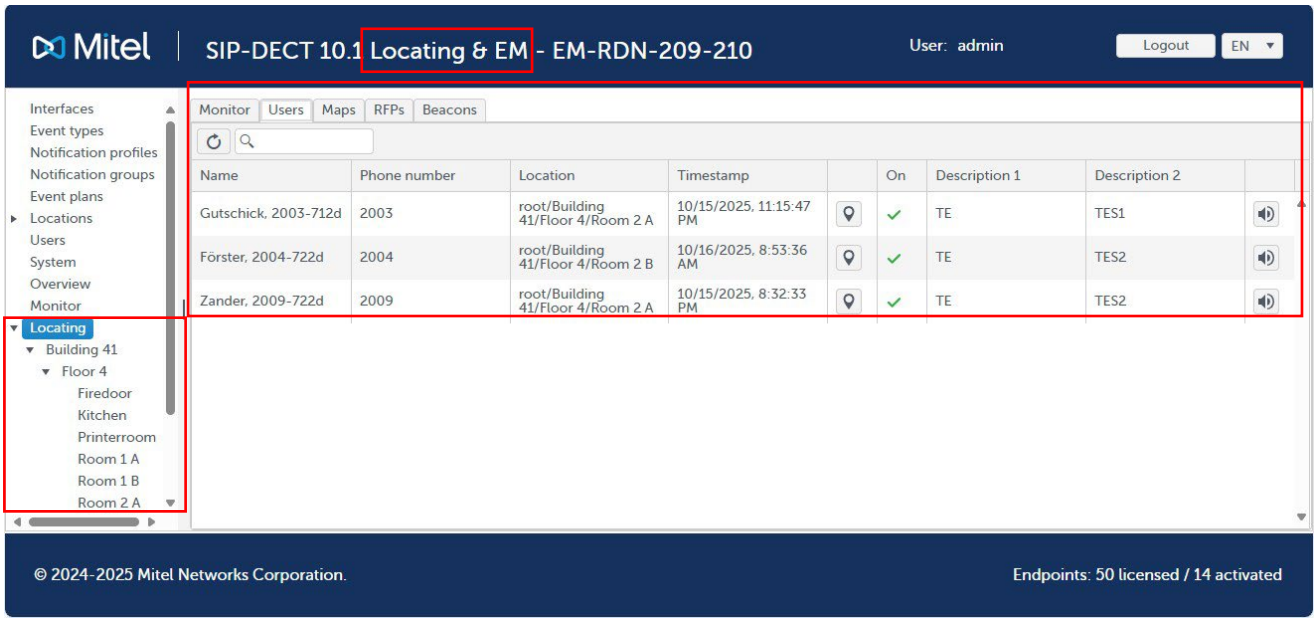
BLE Locating	✓
Event Manager	10.103.37.191

In this case the OMM Status page will show the availability of BLE locating feature.

	Locating		
Number of users allowed to be located	50		*Mitel SIP-DECT BLE Locating License XXX User*
EM Locating application	✓		*Mitel SIP-DECT Locating Server License*
License key	BDWZ9-DZ12B-Z194X-R6FBU-XC2V4		

The Mitel SIP-DECT Locating Server license must be imported into the OMM before the locating functionality is visible on the EM Web GUI.

As soon as the license has been applied and made available to the EM application, the application name in the top bar changes, Locating appears in the navigation bar and allows the access to the locating functionality such as a list of locatable users.



Please note that only locatable users are displayed and that the Mitel SIP-DECT Locating User License is required for them.

As long as users are not to trigger events or receive notifications, they do not need to have been imported from the OMM into the EM and exist as an endpoint in the SIP DECT interface. They still appear in the list of locatable users. This means that no endpoint license is required for these users.

If users have been imported as endpoints but are only to be located without triggering events or receiving notifications, these endpoints can be set to inactive. They are then still listed in the list of locatable users, but are not counted towards the endpoint license.

Supported DECT Phones

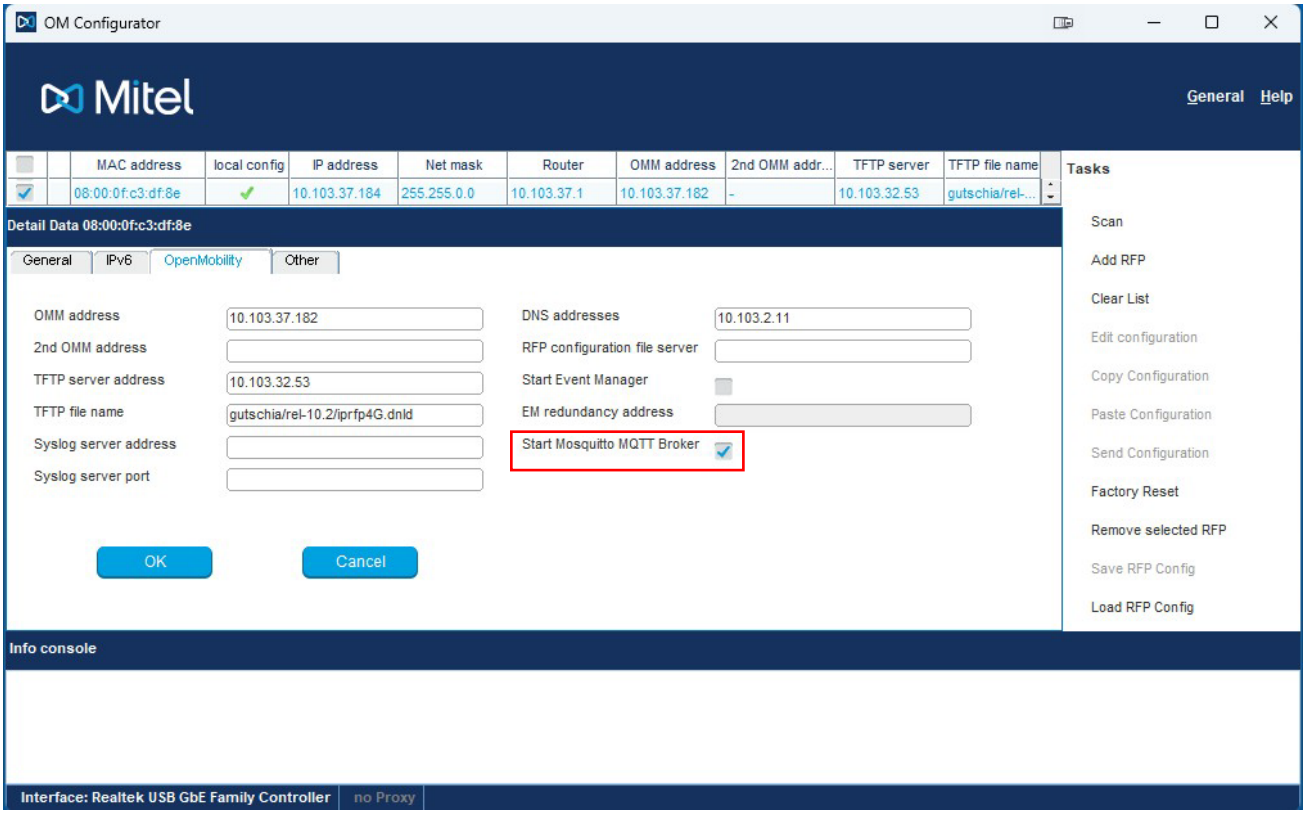
The SIP-DECT Event Manager supports the 700d DECT phone family. The SIP-DECT 600d V2 DECT

phone family is also generally supported. Older generations of the 600d device family or their older SW versions may not support all SIP-DECT messaging features and may therefore have limitations. Please also note the information in the Mitel 600/700 DECT Phone Messaging and Alerting Applications user guide.

Eclipse Mosquitto™ open source MQTT broker on RFP4G

It is possible to start a functionally restricted Eclipse Mosquitto™ open source MQTT broker on a RFP4G. This allows the operation of MQTT in conjunction with the Event Manager and MQTT-capable devices mainly for testing purposes.

The SIP DECT administrator therefor defines in the OMC (OM Configurator) on which RFP the MQTT Broker is started. To do this, the "Start Mosquitto MQTT Broker" flag must be set as shown below.



If the following restrictions are acceptable, the usage in an operational environment is possible.

- Max 150 clients in parallel are supported.
- No support for retained messages (clients which set the retain flag in publish messages will be disconnected).
- QoS 0 is recommended. Please avoid MQTT QoS level 1 and 2 as additional restrictions apply.
- The maximum packet size for single MQTT messages is 4096 Byte (clients sending larger packets will be disconnected). A MQTT message size of ~1200 Byte is recommended to avoid fragmentation and additional CPU and memory load.
- No support for persistent sessions.
- No support for WebSocket connections.
- No support for TLS, only port 1883 supported.

- No client authentication, anonymous access possible.

Due to performance requirements and redundancy configuration of OMM or Event Manager the Broker should not run together with the OMM or the Event Manager on a 4G RFP. If a sufficient number of RFPs is available, the Broker should be activated on a separate RFP.

Additional hints:

The mosquitto broker publishes statistics and usage information under the topic hierarchy '$SYS/broker/#' every 10s.

The tool MQTT-Explorer (https://mqtt-explorer.com/) displays this information by default.

With mosquitto_sub the information can also be retrieved and stored in a file:

```
mosquitto_sub -h <broker-ip-address> -p 1883 -t '$SYS/#' -v
```

The broker logging can be accessed under the topic hierarchy '$SYS/broker/log/#'. The messages are sent here by the broker when the corresponding event occurs. It is not possible to retrieve log messages for events in the past, the broker does not save this information.

Only the log messages of the broker are retrieved with the following command.

```
mosquitto_sub -h <broker-ip-address> -p 1883 -t '$SYS/broker/log/#' -v
```

Note: MQTT-Explorer and mosquitto_sub can be run in parallel on the same broker, the MQTT-Explorer is well suited to display the status and statistics information and mosquitto_sub can be used to record the log output of the broker.

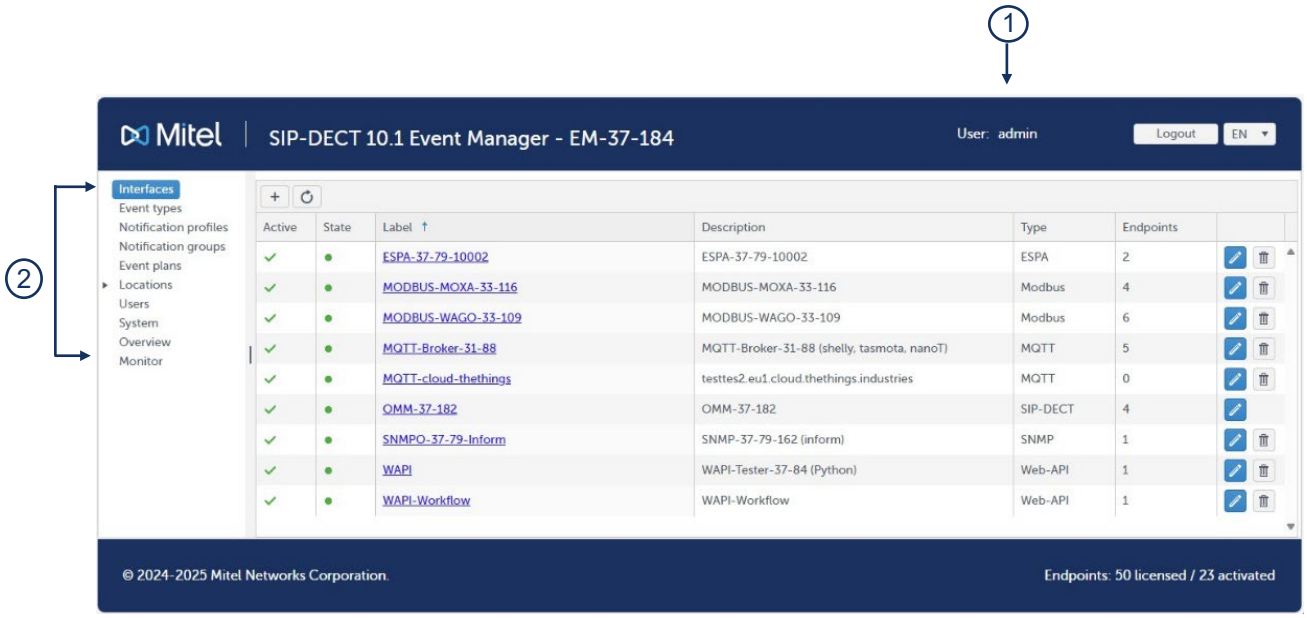
Using the SIP-DECT Event Manager

To take the first practical steps with the SIP-DECT Event Manager as quickly as possible, you can start with the section Quick Start Configuration Guide SIP-DECT Event Manager.

SIP-DECT Event Manager GUI

Admin view

The SIP-DECT Event Manager has its own web administration interface which is available via https://<IP address>:8444. The web interface consists of a series of web pages that are used to configure the various settings of the SIP-DECT Event Manager and can be accessed from any computer or device with a web browser on the same network or via Remote Management (if configured). The web service is implemented as a single-page application (SPA).



1 Login Area

Language Selection

The following languages are available: German, English, French and Spanish. When creating the configuration there are numbers of standard values (e.g. event types) set up in the language selected at this time. These values contained in the configuration are not affected by switching the language.

Use ‘admin’ as the username and password to login for the first time. During login for the first time, the user is asked to change the password.

2 Configuration panes

The SIP-DECT Event Manager includes multiple panes that contain different information about the SIP-DECT Event Manager.

Configuration Pane	Description
Interfaces	The Interfaces pane provides an overview of the status of systems that are connected to the SIP-DECT Event Manager. Interfaces, their

Configuration Pane	Description
	endpoints and interface-specific settings can be managed here.
Event types	The Event types pane allows to create new or change existing Event types. There are **8 default Event types** (‘Man Down’, ‘No Move’, ‘Escape’, ‘SOS-key', ‘System Info’, ‘Locating Alert’, ‘GPS Warning’, and ‘RegDomain Err’) available. These types cannot be deleted. The Event type serves as a kind of filter in an Event plan to control the escalation of an event. Based on the assigned priority, the system can be informed in which order the event should be processed.
Notification profiles	The display and acoustic signaling of an event on the SIP-DECT terminals can be configured within a notification profile.
Notification groups	Endpoints that can receive notifications (e.g. SIP-DECT terminals) can be combined into a notification group. This simplifies the configuration.
Event plans	The Event plans pane allows to create, edit and delete event plans. An Event plan specifies how received events should be handled depending on the location of the originating endpoint. The plan specifies which endpoints should receive notifications and how to react if acknowledgements are not received. An event plan can include one or more event types and one or more locations. It means that the event plan will only be used for events of the configured type and if the originating endpoint belongs to the specified location.
Locations	The Event Manager supports the management of locations to which endpoints are assigned as sources of events. Locations are assigned to event plans too. This allows the location-specific definition of event plans, i.e. it is possible to notify different recipients depending on the location of the sender of an event.
User	The Users pane allows to create, edit and delete users. The default user **admin** cannot be deleted. There are available three user profiles which can be assigned to the users (Configuration, Monitor, Locating).
System	The System pane includes different tabs for naming the system, showing the current software version, for the configuration of a Watchdog and to activate the CloudLink daemon for the remote management. Here can also be performed functions such as Restart, Restart with factory defaults, Export log, Import config, and Export config. The import of SSL certificates is available as well as the configuration of a security level and the used Cipher Suites. In case of activated CloudLink daemon the detailed configuration and displaying the status of it is available.
Overview	The Overview pane presents short summaries of the Event Manager configuration (filterable views regarding event flows, plan executions flows, notification groups, MQTT mappings and interface to endpoint relations).
Monitor	The Monitor pane shows a list of active event handlings and allows the administrator to end a single event or all events. Here is available also a button for triggering Web events (if Web event interface is configured).

Monitor view

The monitor view is the view for users with the permission ‘Monitor’. In this view no configuration is possible.

The only purpose of this view is to display running event flows. The user may cancel one running single event plan or all running event plans. Since release 10.1 there is also the possibility to download log files.

Interfaces

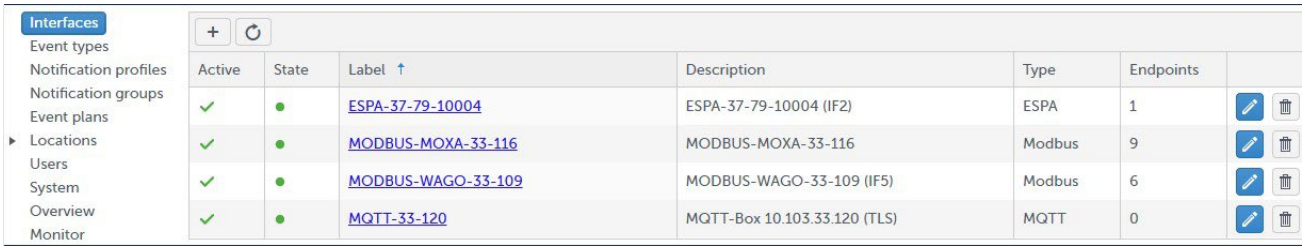
Interfaces connect the SIP-DECT Event Manager to other devices and services. Depending on the type, these interfaces support receiving events or sending notifications, sometimes both.

Depending on the interface type, a certain number of interface instances can be set up until the maximum number of **10** interfaces is reached.

The following types of interfaces can be configured:

Type	Maximum number
SIP-DECT (OMM)	1
ESPA	4
Modbus (e.g. WAGO or MOXA)	4
SNMP	2
MQTT	2
Web-API	4
Web-Event	1
GPS	1

Under the **Interfaces** configuration pane, all configured interfaces are displayed, and can be selected and edited.

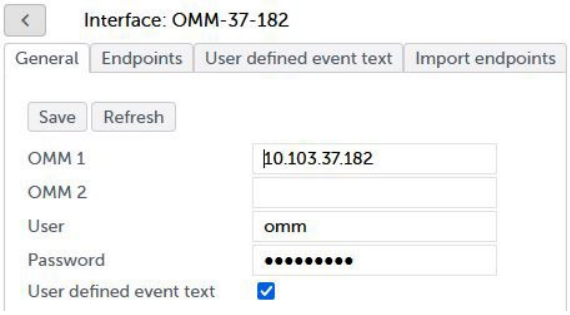


SIP-DECT (OMM) Interface

The SIP-DECT (OMM) interface is already created by default and can only be renamed, but not be deleted. This interface contains the following tabs:

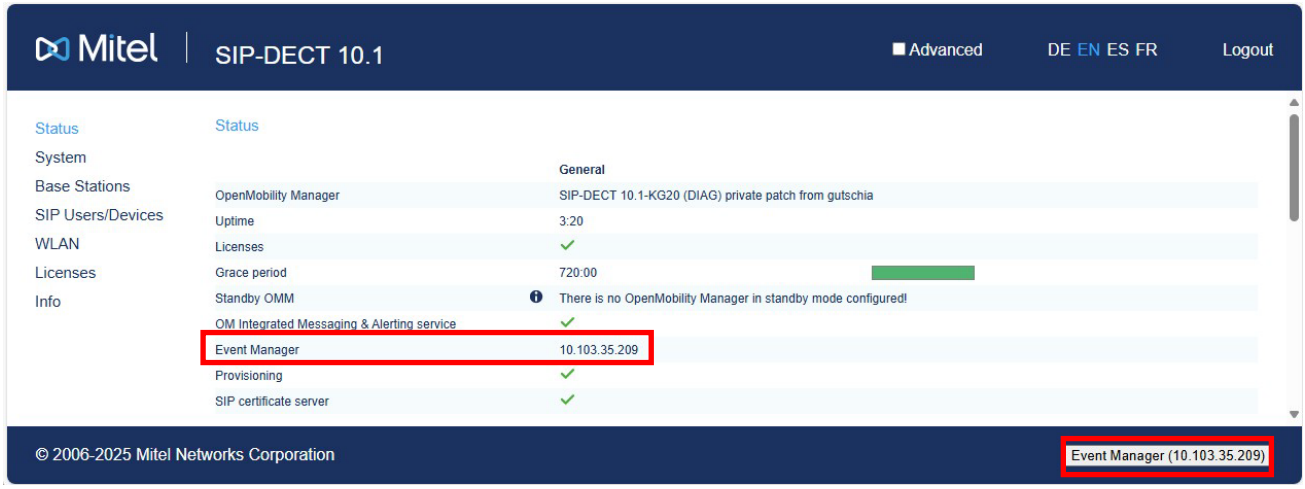
General Tab

The **General** tab is used to configure the OMM IP address(es), user and password. With this configuration the SIP-DECT Event Manager will be able to connect with the OMM. A successful connection is indicated by the interface status turning to green in the interfaces overview tab.



The ‘User defined event text’ box must be selected to take effect the settings under the tab ‘User defined event text’.

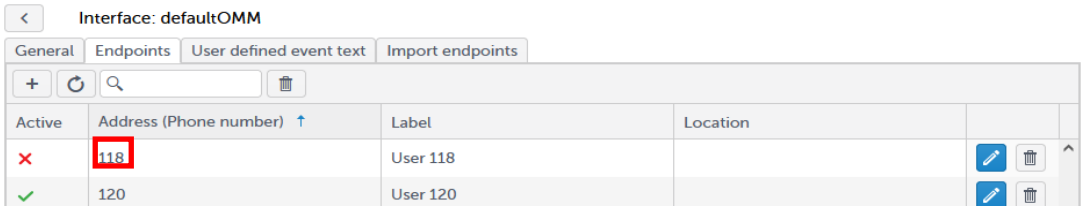
The IP address of a connected Event Manager is available via the OMM web admin on the ‘Status’ page and in the footer of the OMM Web UI.

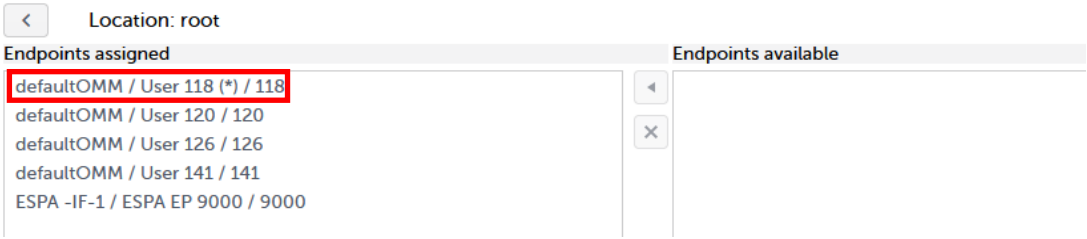


Endpoints Tab

The **Endpoints** tab is used to define the destinations or receivers of messages in the SIP-DECT event. To simplify the setting up of the endpoints on the SIP-DECT interface, these endpoints can be imported via the ‘Import endpoints’ tab.

Please be aware that an endpoint which is not marked as active, cannot be used to trigger an alarm and is not counted as a licensed endpoint. Inactive endpoints are marked with (*) in other configuration panes as shown below.





User defined event text Tab

The **User defined event text** tab is used to customize special types of text to be sent to the DECT phones when an event is handled.

This function allows organizations, agencies, or individuals to create and send messages with specific details or instructions that are relevant for a special situation.

The texts defined in this section only take effect when the checkbox ‘User defined event text’ under tab ‘General’ is selected.

The message text is normally made up of the event type and the location of the originating endpoint. The composition of alarm texts can be flexibly configured for each interface with user defined alarm texts.

The text delivered by the interface during the triggering of the event can be changed before the further editing by replacing individual character strings. The character strings to be replaced should be entered in ‘Text’ and ‘Replaced by’.

Up to four texts can be used for the composition of the final alarm text. A maximum length should be defined for each of these texts. Either a space or a line feed can be used as a spacer between these texts. Since line feeds cannot be displayed on all endpoints, they are automatically replaced with spaces where necessary.

The following texts are available:

- Event type
- Event type short – max 8 characters
- Priority – Priority of the alarm defined by the alarm type
- Originating endpoint (name) – Name of the endpoint at which the alarm has been triggered
- Originating endpoint (address) – Address (e.g. phone number) for the endpoint at which the alarm has been triggered
- Location of originating endpoint – Environment to which the alarm which has been triggered is assigned by the configuration or by DECT locating
- Event phase – The designation of the current escalation phase

- Received text from interface – Permits the use of composed alarm texts based on special interface settings (e.g. ESPA)

Import endpoints Tab

The **Import endpoints** tab allows the automatic import of the SIP-DECT devices configured in the SIP-DECT system as endpoints to the SIP-DECT Event Manager configuration. This function can only be used if a connection has been established between the SIP-DECT Event Manager and the SIP-DECT system (OMM).

If the number of endpoints permitted by the license is exceeded during the import, a warning will be displayed.

Only those endpoints should be imported that are really needed.

The imported endpoints can be deleted under the Endpoints tab.

ESPA Interface

The ESPA interface enables the connection of devices that support data exchange in accordance with the ESPA 4.4.4 protocol. This protocol was defined by the European Selective Paging Manufacturer's Association for controlling radio paging equipment and for connecting fire alarm and light signaling systems.

The SIP-DECT Event Manager supports the ESPA 4.4.4 protocol over IP. This permits the exchange of messages with fire alarm systems, light signaling systems, radio paging equipment and similar systems which also support this interface. An ESPA interface can only operate as an input interface (where the SIP-DECT Event Manager receives messages) and not as an output interface (where the SIP-DECT Event Manager sends messages).

If supported by the other side, the SIP-DECT Event Manager facilitates monitoring of the ESPA connection protocol-wise.

Components are connected directly via TCP/IP byte stream or via RS-232 / IP converter. The SIP-DECT Event Manager acts as a TCP client in an ESPA slave mode.

An ESPA message contains information organized in numbered fields. The following fields are important for configuring the SIP-DECT Event Manager

No.	Designation	ESPA Standard Designation	Remarks
1	Call address	Call Address	16 characters max.
2	Display message	Display Message	128 characters max.
3	Ringtone	Beep coding	
4	Ring type	Call type	
6	Priority	Priority	

Please note: ESPA messages in a wrong format will not be processed. Unknown fields will be ignored. ‘Call address’ (1) and ‘Display message’ (2) must always be present in an ESPA record.

The fields ‘Beep coding’ (3), ‘Call type’ (4), and ‘Priority’ (6) have no direct influence on the notifications to the SIP-DECT phones. They are only used to select the right event type.

The ESPA interface contains the following tabs:

- General
- Endpoints
- User defined event text
- Event assignment
- Simulator/Trace

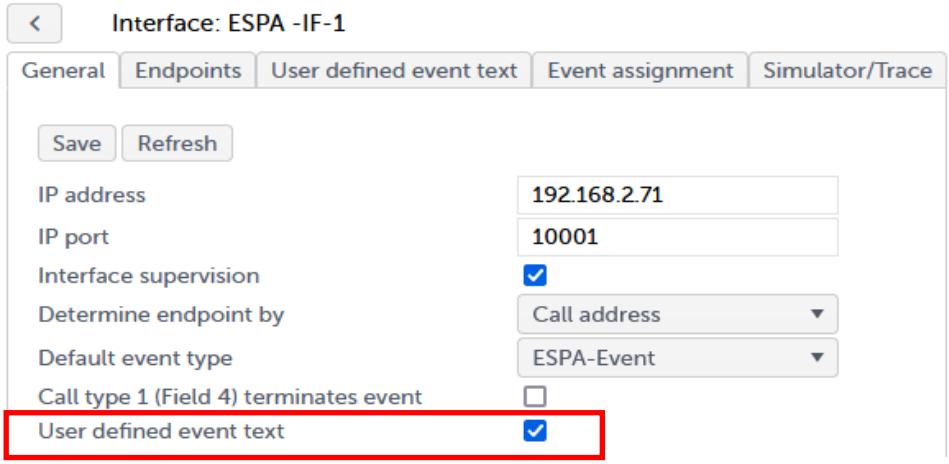
Note: The changes made on the **User defined event text** tab, are only effective if the check box on the **General** tab is selected.

General Tab

The **General** tab allows configuring the basic settings of the ESPA interface. The following settings can be configured:

- **IP address**: IP address to which the SIP-DECT Event Manager should connect to

- **IP port**: The IP port to which the SIP-DECT Event Manager should connect to
- **Interface supervision**: Select this check box if this interface should be supervised.
- **Determine endpoint by**: Select the method for determining the endpoint. Available options are 'Call address' (which is the default setting) and 'Message text'.
- **Default event type**: Select the default event type. A specific event type must be created for it in the Event type section. This default event type is used as fallback if nothing else is defined in the Event assignment tab or if nothing fits to the made configuration.
- **Call type 1 (Field 4) terminates event**: Select this check box to terminate the event.
- **User defined event text**: Select this checkbox if this feature should be used!

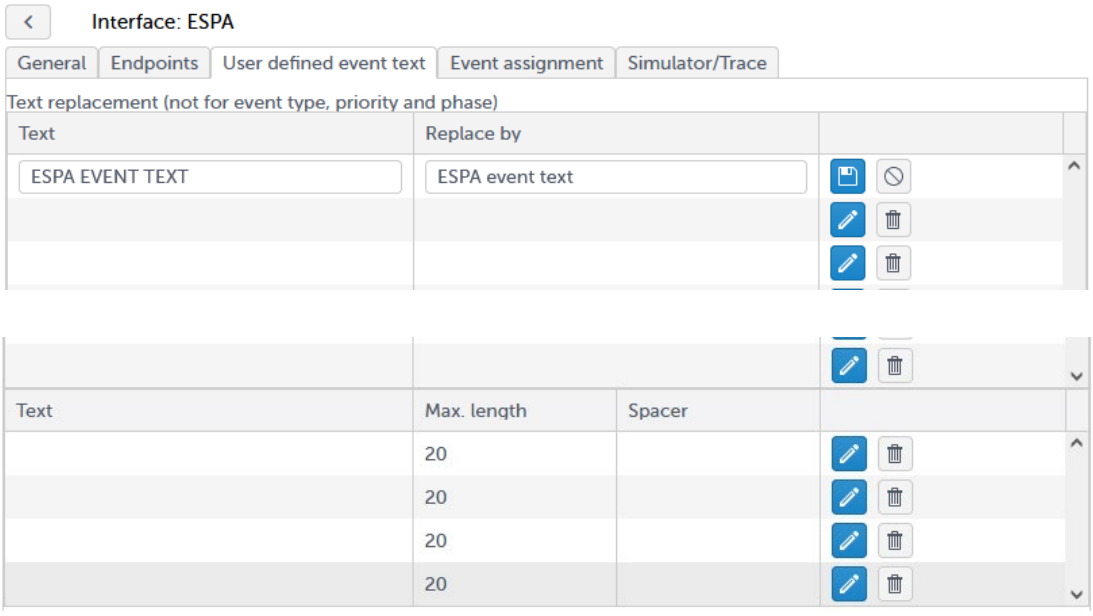


Endpoints Tab

The **Endpoints** tab allows the definition of senders of ESPA messages. The assignment of an endpoint to an ESPA message is done based on the call address. The call address can be determined either by the ESPA field 1 (Call address) or by the ESPA field 2 (Message text). If 'Determine endpoint by: Message Text' is set, the message text must contain only the call address and nothing else.

User defined event text Tab

In the **User defined event text** tab, it is possible to define special content for the notification messages to addressed endpoints (e.g. SIP-DECT terminals). If this feature is not enabled in the **General** tab, the ESPA field 2 (Message text) is used for the notification message. There are two tables available under this tab where a simple text replacement and/or a complete text definition depending on some known parameters is possible.



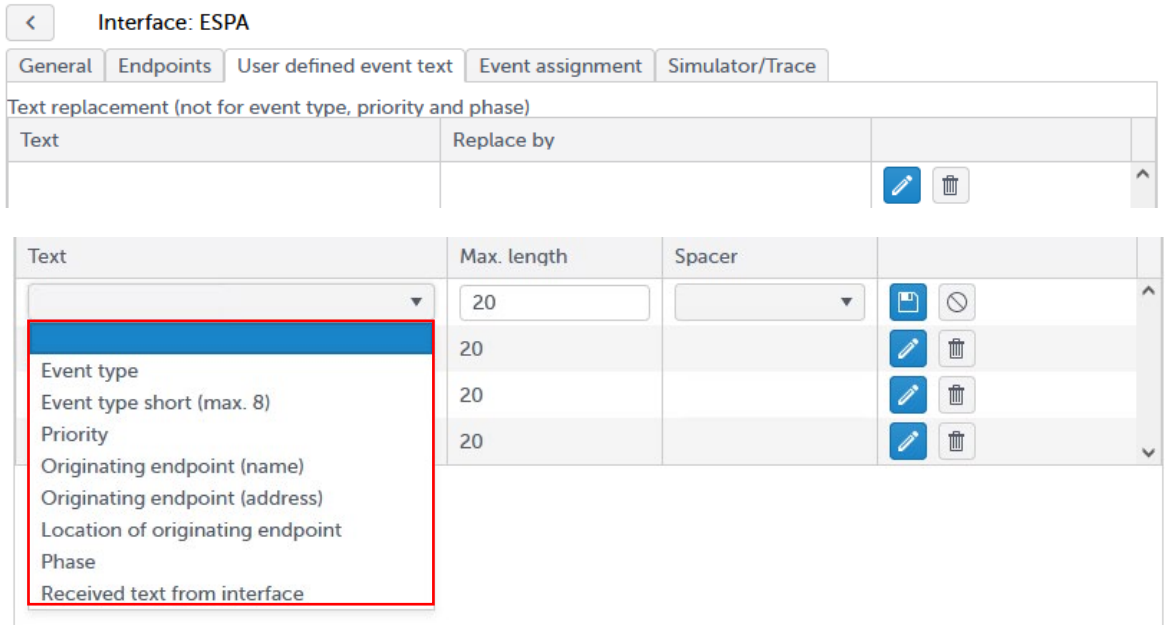
Simple Text replacement

In the table at the top of this tab the received text (field 2) from the ESPA message can be modified.

Text (field 2) of the ESPA message	Replacement rule	Resulting notification text
ESPA EVENT TEXT	ul	ESPA event text

Compose a new event text based on an ESPA message

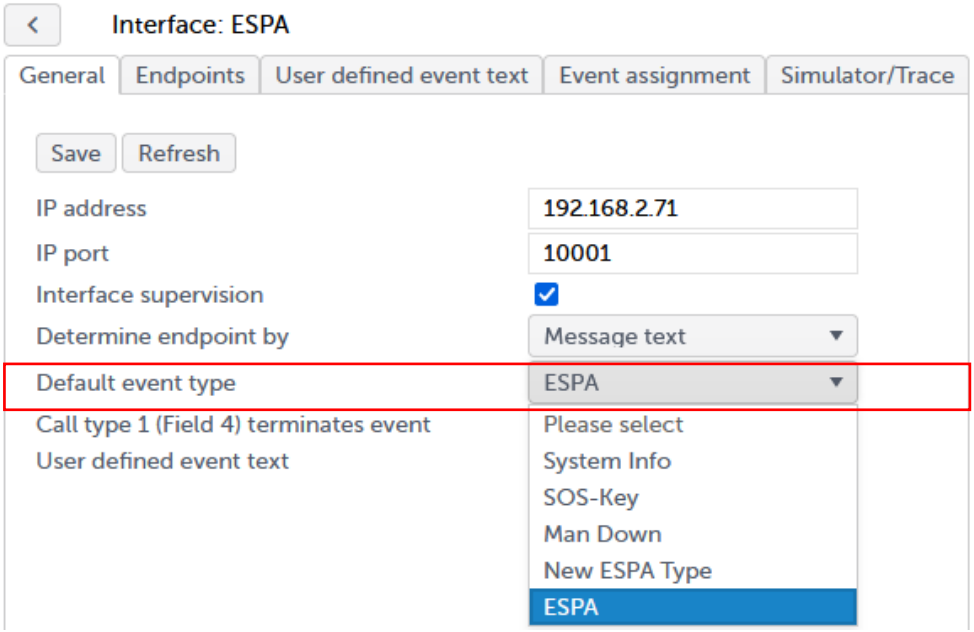
In the table at the bottom of this tab the event text can be recomposed from up to 4 elements. These 4 elements can be selected from 8 different event information elements. These information elements are shown in the following example.



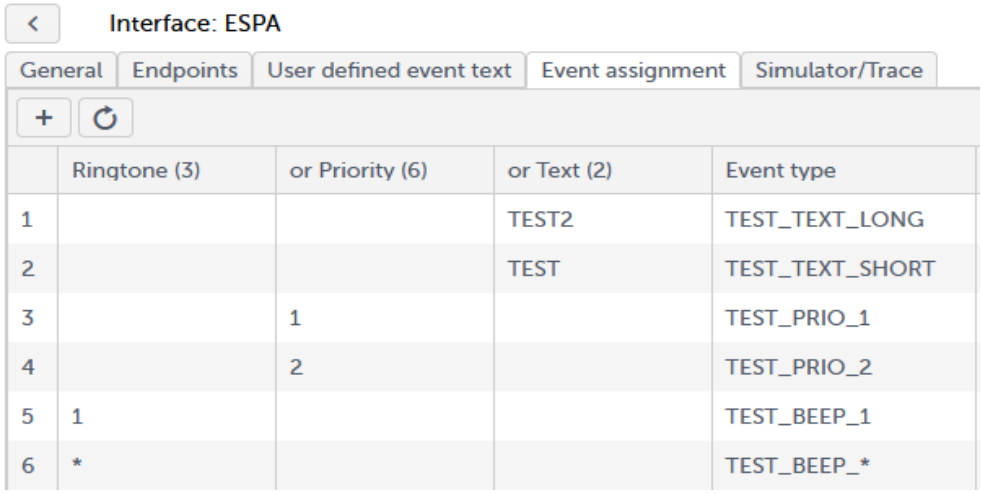
Event assignment Tab

The **Event assignment** tab allows to define the process of designating or assigning specific tasks, roles, or responsibilities to individuals or teams in response to an emergency event. It is a crucial part of coordinating an effective response to emergencies.

An event type is assigned for incoming ESPA messages based on the Ringtone (field 3), Priority (field 6) or Text (field 2). In addition, a Default event type must be configured for non-assigned types in the **General** tab.



Rules can be defined in the **Event assignment** tab of the ESPA interface configuration, as following shown.



	Ringtone (3)	or Priority (6)	or Text (2)	Event type
1			TEST2	TEST_TEXT_LONG
2			TEST	TEST_TEXT_SHORT
3		1		TEST_PRIO_1
4		2		TEST_PRIO_2
5	1			TEST_BEEP_1
6	*			TEST_BEEP_*

Rules are displayed in the order of their creation and are also processed in this order (top down). The first matching rule will be applied. Hence, the more specific rules need to be configured first.

The fields are linked with ‘OR’, not with ‘AND’!

A ‘*’ can be used as a wildcard in the fields ‘Ringtone’ and ‘Priority’. The assignment is then made for all

values used in these fields.

Leading or trailing spaces in the Text field will be removed automatically.

The search for an event will be done in the following order:

1. A search is made for matching values without wildcards.
2. If no such rule applies, the system then searches for wildcards in the ‘Ringtone’ and ‘Priority’ fields.
3. If it is also then not possible to assign an event type, the default event type is used.

For example, a rule with ‘TEST2’ as text is more specific than a rule with text ‘TEST’. To avoid that the ‘TEST’ will always be applied before ‘TEST2’, the rule with text ‘TEST2’ needs to be configured first as shown below.

The following table shows how these rules are applied to some ESPA message input examples.

ESPA message input			Matching rule			Resulting event type	Comment
Ringtone (3)	Priority (6)	Text (2)	Ringtone	Priority	Text		
Any or not provided	Any or not provided	TEST2			TEST2	TEST_TEXT_LONG	Rule 1
Any or not provided	Any or not provided	TEST3			TEST	TEST_TEXT_SHORT	Rule 2
1	1	Hello!		1		TEST_PRIO_1	Rule 3
1	3	Hello!	1			TEST_BEEP_1	Rule 5
Any, except 1	Any (except 1 and 3) or not provided	Hello!	*			TEST_BEEP_*	Rule 6
Not provided	Not provided	Hello!				ESPA	no match, default event type

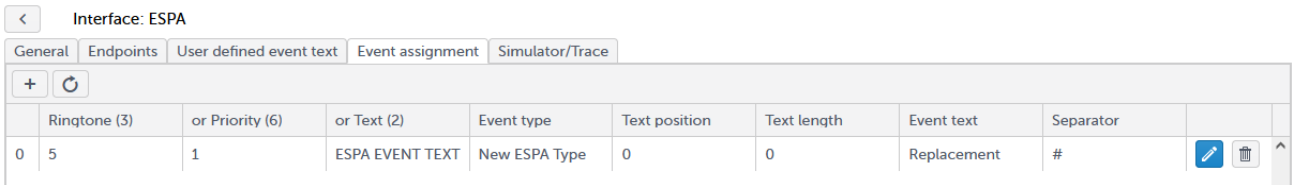
Event Text Replacement

Normally the ‘Message text’ (field 2) of an ESPA message is used as the notification text. Leading and trailing spaces in this text field are not supported and will be removed automatically during the configuration.

If there is an event text defined, then the event text will replace the content of the received ‘Message text’ (field 2) of the ESPA message.

If ‘text position > 0’ is set, then the ‘Message text’ (field 2) of the ESPA message is also included in the notification text starting at the specified text position.

If there is additionally a text length set, then only the specified portion of the ‘Message text’ (field 2) of the ESPA message is also included in the notification text.

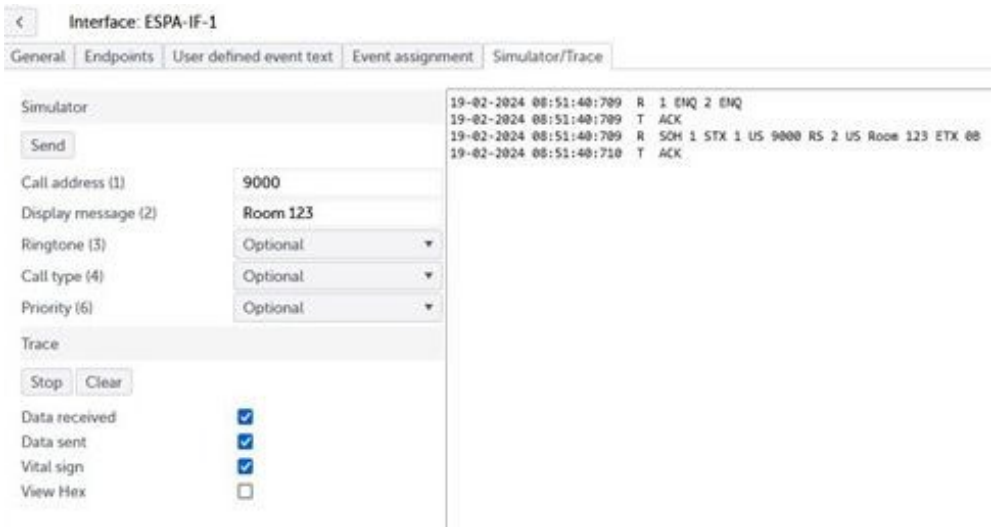


Settings – Text position, Text length and Event text	Resulting notification text
Text (2) / Event type / Text position / Text length / Event text ESPA EVENT TEXT / New ESPA Type / 0 / 0 / Replacement	Replacement
Text (2) / Event type / Text position / Text length / Event text ESPA EVENT TEXT / New ESPA Type / 0 / 0 /	ESPA EVENT TEXT
Text (2) / Event type / Text position / Text length / Event text ESPA EVENT TEXT / New ESPA Type / 1 / 0 / Addition -	Addition - ESPA EVENT TEXT
Text (2) / Event type / Text position / Text length / Event text ESPA EVENT TEXT / New ESPA Type / 6 / 0 / Addition -	Addition - EVENT TEXT
Text (2) / Event type / Text position / Text length / Event text ESPA EVENT TEXT / New ESPA Type / 6 / 5 / Addition -	Addition - EVENT
Text (2) / Event type / Text position / Text length / Event text ESPA EVENT TEXT / New ESPA Type / 6 / 5 /	EVENT

Simulator/Trace Tab

The **Simulator** function can be used to check if a received ESPA message would be escalated correctly. The ESPA interface itself does not need to be running (state: green) for the Simulator function to work. There must only have been created an ESPA endpoint with a location, and in the **General** tab, a Default event type must be selected, and any IP address and port must be configured.

The communication between the SIP-DECT Event Manager and the ESPA interface can be recorded at the protocol level as needed. The **Trace** function can be used to monitor the data sent and received by the ESPA interface. The trace functionality can be started and stopped by the same button.



Modbus interface

The Modbus interface enables the connection of devices e.g. WAGO or MOXA which provides input ports (e.g. buttons or switches) and output ports (e.g. lights) via Modbus-TCP protocol. The Modbus protocol is a client / server data protocol in the application layer of the OSI model which was originally published by Modicon (now Schneider Electric) in 1979 for use with programmable logic controllers via RS232/RS485 interfaces (Modbus-RTU). For data transmission over Ethernet the protocol was adapted to Modbus-TCP. Meanwhile Modbus has become a de facto standard communication protocol for communication between industrial electronic devices in a wide range of buses and networks.

Reading digital input ports and setting digital output ports of Modbus-TCP devices is supported by the Event Manager.

The following devices have been approved for correct interoperability with the Event Manager:

- WAGO I/O System 750 (“Fieldbus Coupler Modbus TCP 4th generation” Item no. 750-362)
- MOXA ioLogik E1200 Series (ioLogik E1212)

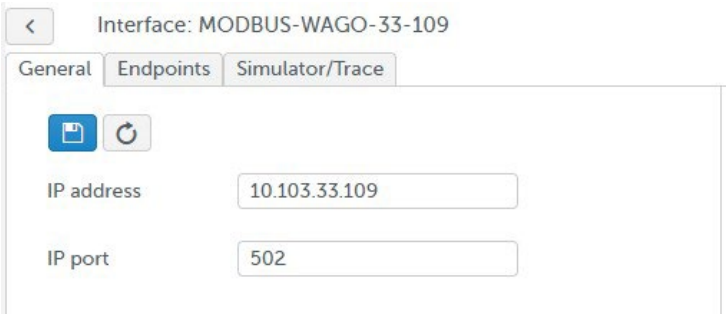
Analog inputs and outputs and other sensor ports are not supported by the Event Manager.

Note: Functionality cannot be guaranteed with other devices and must be checked separately before use. The following conditions must be observed.

- Only digital inputs/outputs supported (no analog inputs/outputs or other sensors)
- IO addresses must not be remapped by device configuration, Event Manager only supports address range starting with address 1 for input/output ports.

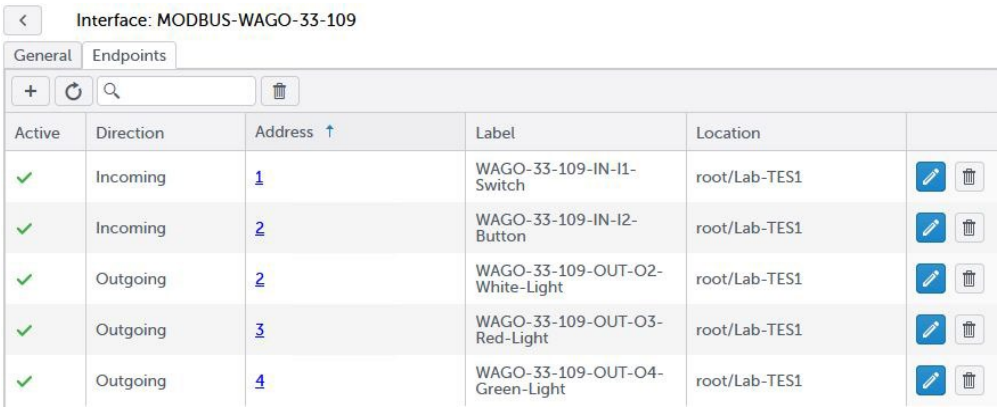
General Tab

The **General** tab is used for configuration of the IP address and port of the Modbus-TCP device which is connected through the interface.

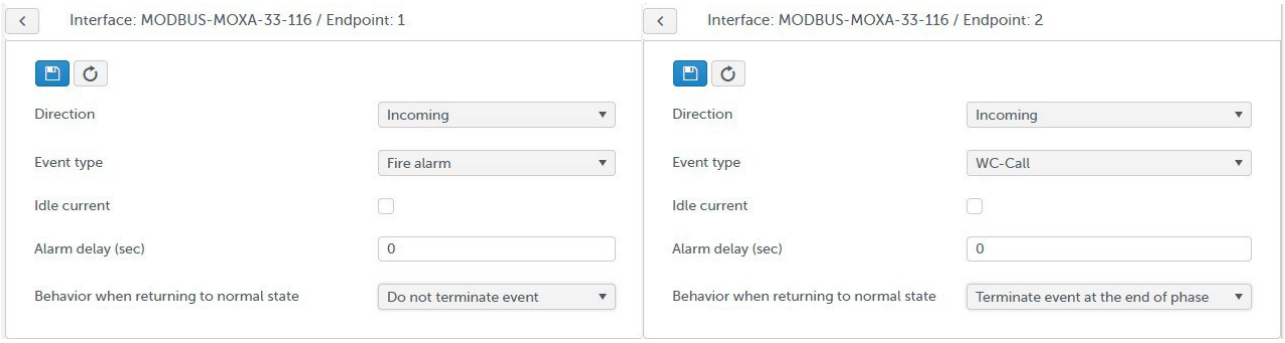


Endpoints Tab

The Endpoints tab is used for configuration of incoming and outgoing endpoints. Incoming endpoints correspond to digital inputs of Modbus-TCP devices and outgoing endpoints correspond to digital outputs of Modbus-TCP devices. For WAGO devices the incoming ports 1-256 are valid addresses, for MOXA only the addresses 1-16.



In the Endpoints configuration (reached by the link in the overview) some special settings for the endpoint can be configured. Mandatory are ‘direction’ and ‘event type’, optionally some special settings can be configured: ‘Idle current’ or ‘Working current’ is used on the connected device, a ‘Alarm delay in seconds’ and the ‘Behavior when returning to normal state’ (not terminate, terminate immediately or terminate at the end of the current alarm phase). For outgoing endpoints can be configured no special settings.



Simulator/Trace Tab

The **Simulator/Trace** tab is used for simulation of the Modbus interface endpoints and for tracing changes on input/output ports. Each time the tab will be opened the trace window will show TCP/IP related connection information and the simulation window will show the actual status of the configured ports. By pressing the “Show all inputs” button the state of all inputs between address 1 and highest configured incoming endpoint address is shown. It is not recommended to open more than one browser window with active Simulator/Trace tab. Only one session is handled by the system.

For configured incoming endpoints a small button is drawn beside each input state. If such button is pressed, the event configured for this endpoint will be generated and processed with defined event plan, Please note that the configured endpoint attributes “Alarm delay”, “Idle Current” and “Behavior when returning to normal state” don’t apply if this button is pressed, the configured event will be generated immediately.

If needed the executed event plan can be canceled via Monitor section of the Event Manager web frontend.

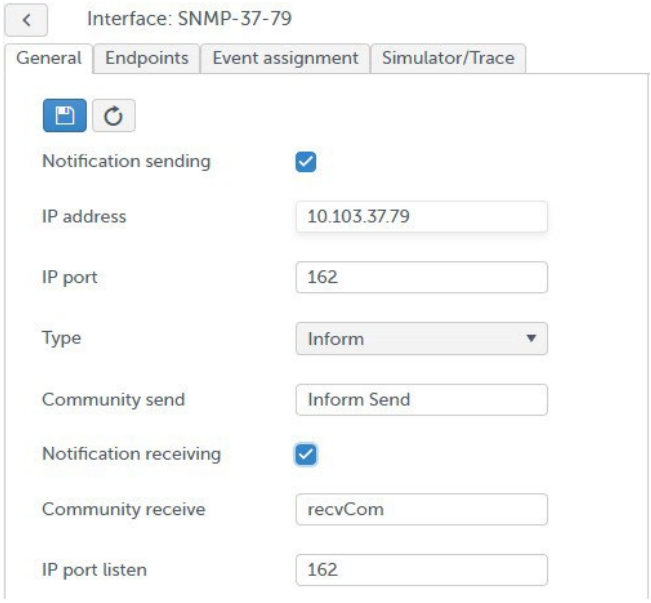
In the Outputs part of the **Simulator/Trace** tab the activity on output port 1 (in this example there is a light connected) is visible and in the trace part of the tab the handled trigger event at the incoming port 1 (triggered by the switch connected physically to this port or by pressing the button 1 in the Inputs part) is documented.

For simulation of the Modbus interface without a connection to a physical device it is possible to configure the interface with local host IP address (127.0.0.1).

SNMP interface

General Information

The SNMP interface enables the Event Manager to send and receive SNMP notifications to and from configured IP addresses with correct community strings. Sent notifications as well as received ones may only be Traps or Inform-Requests. Only SNMP v2c is supported for sending notifications while SNMP v1 and SNMP v2c is supported for receiving notifications.



Notification sending

In order to send notifications, “IP address”, “IP port”, “Type” and “Community send” need to be correctly configured. Should the selected SNMP interface only send notifications, you may uncheck “Notification receiving”.

“IP address” and “IP port” determine where a notification shall be sent to.

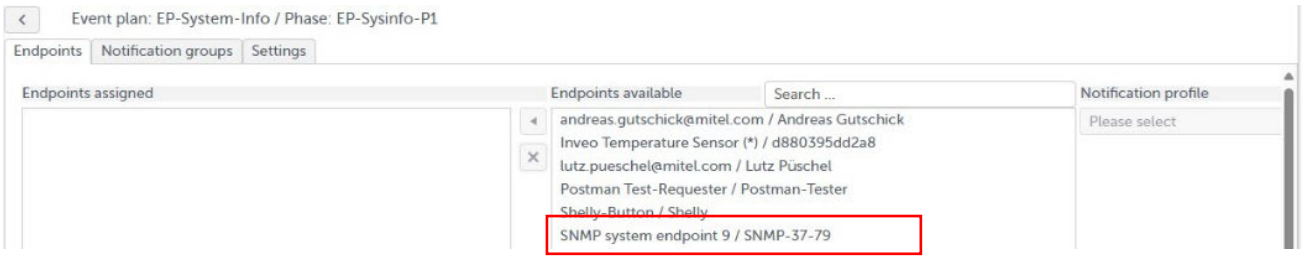
“Type” tells the interface whether to send Traps or Inform-Requests. Traps are notifications that are sent once without the Event Manager checking whether the configured recipient has received them. In the case of Inform-Requests however, the Event Manager waits for a correct Get-Response from the target. Should a correct Get-Response not be received after 5 seconds, the Inform-Request will be resent. The Event Manager will only resend an Inform-Request once (so twice total) before timing out.

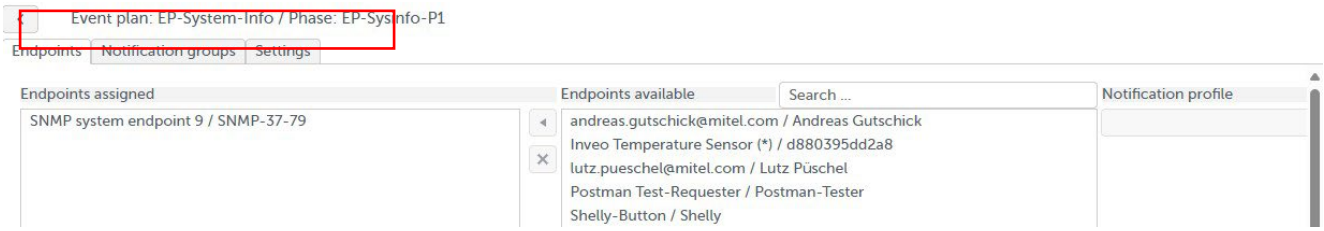
“Community send” sets the sent notifications community string. This community string must match whatever community string the configured recipient has configured. Otherwise, the recipient will not process our sent notification.

Once an SNMP interface has been added, a system endpoint for sending notifications is automatically created. This endpoint will count towards the licensed endpoint count for as long as the checkbox “Notification

sending" in that same SNMP interface remains checked. This endpoint cannot be edited or deleted in any way and can't be assigned to a location.

To give the SNMP interface the ability to send notifications, you need to add this system endpoint into an event plan's phase like any other notification endpoint. Once that phase is activated, the corresponding SNMP interface will send a matching notification to its configured recipient.





Interface Status Change

Should the event plan be triggered by the predefined event type "System Info", the notification will contain data about the interface that triggered it and the current status of it. A "System Info" event is triggered by any interface when its status changes. This event is always triggered in the location "root". If an SNMP interface is supposed to send notifications about interface status changes, an event plan handling the predefined "System Info" should be configured in that location with a phase containing the SNMP system endpoint as an assigned endpoint. Modifying the event type "System Info" has no influence on this functionality.

Notification name	Data field name	Object Identifier (OID)	Comment
interfaceStatusChange	---	.1.3.6.1.4.1.1027.4.1.1337.0.4	the snmpTrapOID value
	interfaceType	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.4	the interface's type
	interfaceLabel	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.2	the interface's name
	interfaceState	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.6	the state the interface has now changed to
	InterfaceDescription	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.3	description of the interface

Event Plan Processing

When a phase with an SNMP endpoint is activated, the corresponding SNMP interface will send a notification to the configured target. This notification will contain a notification ID, the event text, data about what triggered the plan and information on the triggered plan and phase. Once the phase has ended by any means, an additional notification with a matching notification ID will be sent to the target, informing about the end of the phase. This notification does not contain the reason for ending the event phase and/or event plan. The current implementation is offered for evaluation of use cases. Accordingly, this functionality may be further developed and may be subject to technical changes in future software updates.

Notification name	Data field name	Object Identifier (OID)	Comment
activateEventPhase	---	.1.3.6.1.4.1.1027.4.1.1337.0.5	the snmpTrapOID value exact same fields as deactivateEventPhase
deactivateEventPhase	---	.1.3.6.1.4.1.1027.4.1.1337.0.6	the snmpTrapOID value exact same fields as activateEventPhase
	trapEventID	.1.3.6.1.4.1.1027.4.1.1337.0.3.1	this ID matches in corresponding activate and deactivate notifications
	trapEventText	.1.3.6.1.4.1.1027.4.1.1337.0.3.2	the event text
	locationLabel	.1.3.6.1.4.1.1027.4.1.1337.2.1.3.1.2	location where the event plan was triggered
	endpointLabel	.1.3.6.1.4.1.1027.4.1.1337.4.1.3.1.5	name of the endpoint that triggered the event
	endpointCallNumber	.1.3.6.1.4.1.1027.4.1.1337.4.1.3.1.3	call number of the endpoint that triggered the event
	eventTypeLabel	.1.3.6.1.4.1.1027.4.1.1337.3.1.3.1.2	name of the event type
	eventPlanLabel	.1.3.6.1.4.1.1027.4.1.1337.6.1.3.1.2	name of the event plan
	phaseLabel	.1.3.6.1.4.1.1027.4.1.1337.6.1.4.1.3.1.2	name of the phase
	phaseDuration	.1.3.6.1.4.1.1027.4.1.1337.6.1.4.1.3.1.6	Duration of phase in seconds

coldStart Notification

Once an SNMP interface is correctly configured, it will send a coldStart notification to its configured target. This notification will be sent every time the SNMP interface is modified in such a way that it can work correctly or when it is activated after being switched off. This notification will also be sent when the event manager starts or gets rebooted, if they are configured correctly. These coldStart notifications make the interface visible to SNMP management systems. They are however only supposed to inform the recipient that the SNMP interface itself is configured correctly and ready to send notifications. They do not yield concrete information about the state of the event manager itself or other interfaces. Furthermore, the Event Manager does not send warmStart notifications, even if the configuration of the interface did not change.

Additional Notification Fields

Each notification contains MIB undefined data fields in addition to their defined ones. These notification fields include information about the Event Manager itself and data that is too specific for the more generic notification type. They are appended after the MIB defined data fields.

Notification name	Data field name	Object Identifier (OID)	Comment
Additional fields	---	---	data fields that get appended to notifications after their MIB defined data types
	sysName	.1.3.6.1.2.1.1.3	appended to all notifications, EVP's name
	systemIPAddress	.1.3.6.1.4.1.1027.4.1.1337.10.3	appended to all notifications, EVP's IP address
	systemMACAddress	.1.3.6.1.4.1.1027.4.1.1337.10.4	appended to all notifications, EVP's MAC address
	systemVersion	.1.3.6.1.4.1.1027.4.1.1337.10.2	appended to all notifications, EVP's version number
	snmpTrapEnterprise	.1.3.6.1.6.3.1.1.4.3	always last data field, contains Mitel's Enterprise OID

Management Information Base

In order to interpret these messages and their data fields correctly, two MIB files are supplied together with the Event Manager. The first Management Information Base (MIB) is Mitel's root MIB file (Mitel-MIB.mib). It is necessary for the second MIB, the Mitel-EVP-MIB.mib, to work. Both .mib files together contain all the proprietary information that an SNMP agent needs to correctly interpret the specific data and notifications of the Event Manager.

Other RFC defined MIB files that the Event Manager utilizes are SNMPv2-SMI (RFC 2578), SNMPv2-TC (RFC 2579), SNMPv2-CONF (RFC-2580) and SNMPv2-MIB (RFC 3418).

Receiving notifications

In order to receive and process SNMP notifications, "Notification receiving" must be checked and the fields "Community receive" and "IP port listen" need to be configured. Should this interface only receive notifications, "Notification sending" may be unchecked.

"Community receive" configures the community string all received notifications need to have in order to be processed. In case of wrong community strings the Event Manager will ignore the associated notification and no further processing will take place.

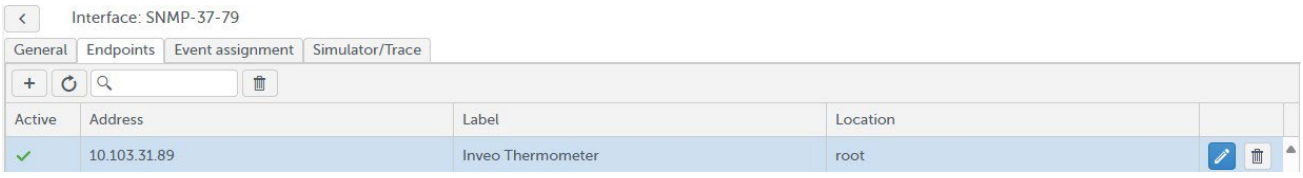
"IP port listen" is the port on which this SNMP interface will listen to Traps/Inform-Requests. Should the SNMP interface not be able to open said port for any reason, its status will change to "Inactive" (red). If that is the case, please select a different listening port. Be aware that two different SNMP interfaces may not use the same listening port!

The configured receiving port(s) must be added to the firewall settings if the EM is running on a Linux server installation e.g. like with the following command:

```
firewall-cmd --zone=public --permanent --add-port=162/udp.
```

The Event Manager will process received notifications (Traps and Inform-Requests) in order to trigger events. Received Inform-Requests will be answered with correct Get-Responses and received Traps will not be answered but only processed. Any other type of Request or PDU will be ignored, they won't trigger events, and will go unanswered.

In order for notifications to be processed into events, a receiving endpoint has to be configured. Only notifications from configured and active endpoints will be processed. Only in case a notification is received from a configured and active endpoint with a correct community string an event will be triggered in the endpoint's assigned location. The triggered event type is determined by the first matching event assignment. Should no valid Event assignment be found, no event will be triggered.



Field name	Explanation
Nr.	The order in which the event assignments have been created, with the lowest number being the earliest created. The first matching event assignment triggers the corresponding event, starting from the lowest number.
Label	The name of this event assignment.
Object Identifier	The object identifier (OID) this event assignment corresponds to. Should a received SNMP notification contain a field with this **exact** OID or should its 2nd field be snmpTrapOID (defined: SNMPv2-MIB) and contain that **exact** OID as its value, this event assignment is chosen and its corresponding event will be triggered in the receiving endpoint's location.
Ignore Indices	The amount of OID indices from the end (right) that will be ignored on **incoming** notification's Object Identifiers. The shortened received OID must still **exactly** match the configured OID in the field "Object Identifier".
Event type	The event type to be triggered if this event assignment is selected.
Re-trigger event timeout	The amount of time an event will **NOT** be triggered again by the same endpoint should this event assignment be chosen. This is especially useful if an SNMP notification sender sends way too many SNMP notifications in a short amount of time. All timeouts will reset if the corresponding interface is disabled, enabled or changed in any way.
Units	A short text which is appended to the defined OIDs interpreted data. Matches the UNITS clause inside MIB-definitions.
Display-Hint	Select how the defined OIDs value is supposed to be displayed inside the generated event text. Values that would lead to useless results are discarded upon event text generation. Matches the DISPLAY-HINT clause inside MIB-definitions but has been simplified to a dropdown menu. It is recommended to be left on "Automatic" unless you are 100% sure about what value you will receive after this OID. "Text" = 'a'; "Decimal" = 'd'; "Decimal with decimal places: X" = 'd-X'

A valid event assignment is determined by trying to match its configured OID with all received OIDs as well as the OID inside the predefined snmpTrapOID value field. This is the 2nd field in any SNMP v2c notification with the OID .1.3.6.1.6.3.1.1.4.1(.0). The first matching event assignment will determine the triggered event and the first matching OID inside the received notification will have its value displayed inside the event text.

The event text contains the triggered event type, the endpoint that triggered it and its address, the chosen event assignment's label and the interpreted value behind the event assignment's "Object Identifier"-field according to its "Display-Hint"-field with the "Units"-field simply appended.

Should the "Object Identifier" field be a snmpTrapOID, the event text will indicate that the received value was a "TRAP TYPE" instead of the interpreted value.

Here are some examples of how event assignments are chosen to more easily visualize them.

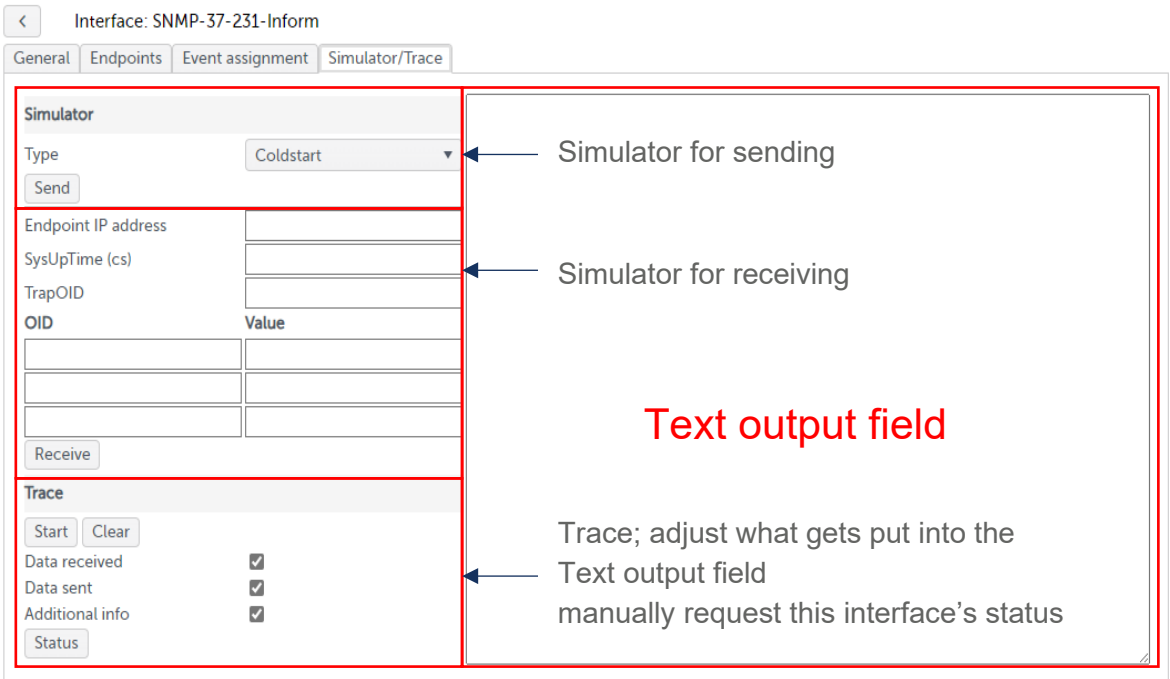
Received OIDs	Received values	Event assignment	What gets checked *)	Final result
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8	37652723 .1.3.6.1.4.5.5.2.4 "Example Text"	OID: .1.3.6.1.2.1.1.3 Ignore indices: 0	.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 .7.6.4.12.5.9.8.8	• No exact match • No event trigger • The next event assignment will be tried
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8	37652723 .1.3.6.1.4.5.5.2.4 "Example Text"	OID: .1.3.6.1.2.1.1.3 Ignore indices: 1	.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 .7.6.4.12.5.9.8.8	• Exact match because 1 index ignored • Event trigger!
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0	37652723 .1.3.6.1.4.5.5.2.4	OID: .1.3.6.1.2.1.1.3 Ignore indices: 2	.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0	• No exact match • No event trigger

Received OIDs	Received values	Event assignment	What gets checked *)	Final result
.7.6.4.12.5.9.8.8	"Example Text"		.1.3.6.1.4.5.5.2.4 .7.6.4.12.5.9.8.8	• The next event assignment will be tried
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8	37652723 .1.3.6.1.4.5.5.2.4 "Example Text"	OID: .7.6.4.12.5.9.8.8 Ignore indices: 0	.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 **.7.6.4.12.5.9.8.8**	• Exact match • Event trigger!
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8 .1.3.6.1.2.1.1.4.0	37652723 .1.3.6.1.4.5.5.2.4 "Example Text" 50	OID: .1.3.6.1.2.1.1 Ignore indices: 2	**.1.3.6.1.2.1.1**.3.0 .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 .7.6.4.12.5.9.8.8 .1.3.6.1.2.1.1.4.0	• Exact match • The first matching OID's value will be used for the event text • Event trigger

*) Red numbers inside received OIDs are what gets compared with the event assignment OID. Black numbers inside received OIDs are ignored when trying to compare with the event assignment OID. Underlined OIDs are matching with the event assignment OID. Bold underlined OIDs are the ones used for the event text.

Simulator/Trace

The Simulator/Trace tab is used to simulate receiving and sending traps.

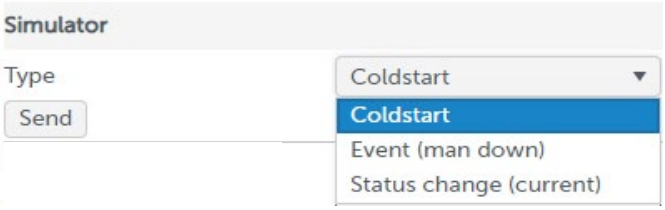


The Trace is used to display what the SNMP interface is sending, receiving as well as other information related to internal activities. "Start" starts the trace output and gets replaced by "Stop", which in turn stops the trace output. "Clear" clears the entire trace output. "Status" prints this SNMP interface's status into the text output field. The checkboxes "Data received", "Data sent" and "Additional info" decide what information is automatically printed into the text output field if the trace has been started. "Data received" enables showing received traps and info relating to it, "Data sent" enables showing sent traps and info relating to it and "Additional info" enables showing info relating to how data is processed and what the result was. Error

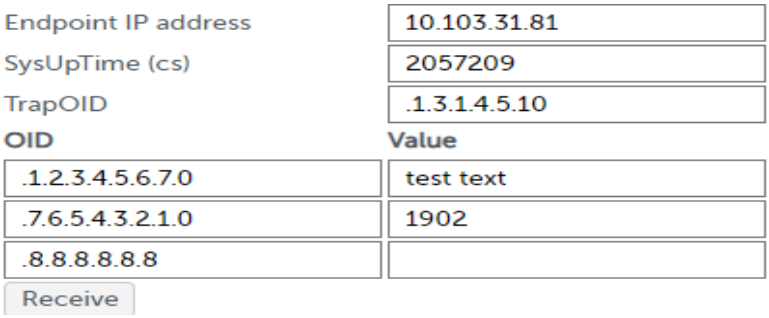
messages are always printed, regardless of which checkboxes are enabled or disabled, so long as the trace has been started.

The Simulator allows you to force the SNMP interface to send predefined traps with the interface's configuration and allows you to test what happens if the SNMP interface receives a customizable trap.

In order to send a predefined trap, select the type of trap this interface is supposed to send and then press "Send" button. The Event Manager will then send that trap type in accord to its own configuration. Currently, the Event Manager will only send coldStart, event related and status change traps.



In order to see what the Event Manager sends, start the trace and check "Data sent". This is useful to test if this SNMP interface is correctly configured for sending traps as well as to check if the trap receiver outside the Event Manager handles traps correctly. Data sent by the Simulator is in the correct format, but data itself may or may not be correct.



The Simulator may also be used in order to simulate receiving a trap to test if the "Event assignment" and "Endpoints" have been done correctly.

Firstly, you need to enter an IP address from which the trap was supposedly sent from.

Secondly, the mandatory SNMP fields "SysUpTime" and "snmpTrapOID" need a valid centisecond value and a correctly formatted OID respectively. The OIDs do not need to be real or MIB defined.

Lastly, you may add up to 3 additional OID-value pairs to the simulated trap. Simply write a real or imagined OID into the left column and a corresponding value into the right column.

When pressing the button "Receive", this interface will generate a trap with the given values (if possible) and send it to itself. In order to see this generated trap, start the trace and check "Data receiving". The output window will display the generated trap as well as the result of processing this trap. Should an event be triggered, an event plan must exist in the correct location in order to catch it and be triggered.

MQTT interface

The MQTT interface connects the Event Manager with an MQTT broker. The interface allows the subscription of user defined topics to the MQTT broker to receive messages from IoT devices that publish their events to this broker. The Event Manager process MQTT messages received from the MQTT broker and trigger events if a match of a user defined condition on an assigned topic is found in the Event Manager configuration. The interface is also able to publish messages to the MQTT broker generated by the Event Manager notification mechanism to trigger actions on other IoT devices that are connected to the same MQTT broker. Up to two MQTT interfaces may be configured.

Only in secure inhouse environments (due to the lack of TLS support and user authentication) an internal MQTT broker may be run as internal application on a dedicated RFP4G of the SIP-DECT system. The performance of this internal broker is sufficient for QoS 0 and usual IoT device traffic (short messages every few seconds), the use of QoS 1and 2 is not recommended, as there is a higher risk for dropped messages by the broker application in case of high load situations. Therefor the broker configuration is limited by default.

General Tab

The **General** tab allows configuring the basic settings of the MQTT interface. The following settings can be configured:

- **IP address**: IP address of MQTT broker
- **IP port**: IP port of the MQTT broker (default: 1883)
- **User**: username configured on the broker
- **Password**: password of the user configured on the broker
- **Use TLS**: set this, if TLS should be used as protocol (default IP port: 8883)
- **Validate certificates**: activate, if server certificates of the broker should be validated
- **User defined event text**: Select this check box if ‘User defined event text’ should be used.

When the correct settings have been configured, the Event Manager will connect to the MQTT broker.

Please note: *The internal MQTT broker does not support TLS and is limited by the default configuration.*

Please note: *If “Use TLS” is not activated, only unencrypted connections without authentication (usually Port 1883 is used by broker for such connection). Eventually the broker setup must be changed to allow such connections. It is not recommended to setup connections to a MQTT broker outside the LAN due to unencrypted data transfer.*

Endpoints Tab

The **Endpoints** tab allows the creation of those IoT devices which shall interact with the Event Manager via the MQTT broker.

User defined event text Tab

In the **User defined event text** tab, it is possible to define special content for the notification messages to addressed endpoints (e.g. SIP-DECT terminals). If this feature is not enabled in the **General** tab, the notification text generated by the MQTT interface consist of the endpoint label and the event type description (or the event type name if the label is empty) and will be used for the notification message. Usually the MQTT payload from IoT devices is not intended to be readable for humans. Therefor this setting and configuration can be used to extract special parts of the received messages and to generate more readable notifications for the receiving SIP-DECT endpoints.

Topics Tab

The **Topics** tab allows the creation of topics that the Event Manager shall subscribe at the MQTT broker or which shall be used on publish for notifications. In the column 'Type' can be selected if the topic is to be used for subscription on the MQTT broker or for publish messages on Event Manager notifications. Each topic must be assigned to a previously created MQTT endpoint. It is possible to configure several topics for an endpoint. All topics must be unique for one interface, it is not possible to create a second entry with the same topic assigned to another endpoint.

MQTT in general allows the use of single level ('+') and multilevel ('#') wildcards in topics on subscription to a broker.

On the Event Manager it is possible to configure any text string as topic including wildcards.
But it is in the responsibility of the Event Manager administrator to configure only valid topic(s) matching the complete topic(s) on which a specific device will publish its data.
A mapping of MQTT messages resulting from a subscribed topic with wildcards is not possible as the received topic is different from the subscribed topic.

Temporary it might be useful to configure a wildcard topic for a specific device to get knowledge about topics and payloads that a specific device is publishing in Event Manager trace (a trace window in the GUI is planned but currently not yet available).

If used at all it is strongly recommended to restrict a wildcard topic to a specific device, otherwise the Event Manager might be flooded by MQTT messages from a lot of devices and get unstable if there are many devices connected to the MQTT broker.

Subscribe mapping Tab

The **Subscribe mapping** tab allows the configuration of mappings for received payloads of the MQTT messages to event types. For each MQTT topic can be added one or more mappings with a condition for the payload. A condition is used to decide whether an event trigger shall be generated or not. Different conditions for the same MQTT topic are used to generating different event triggers on different MQTT payload content.

On reception of a MQTT message at first the topic of the message must match a configured topic in the Event Manager (which must not be disabled in the configuration). Additionally, there must exist a 'Subscribe mapping' for this topic which contains a condition to be used for checking the payload of the MQTT message. The assigned event type will only be triggered if the condition match and is not waiting for leaving the configured hysteresis range or a retrigger event timeout.

On reception of a MQTT message all configured conditions mapped to the received topic will be checked. If more than one condition matches the received message also more than one events might be triggered in case of only one received MQTT message.

Depending the configuration of a 'json_key' for a condition either the json value of the json attribute specified by the key or the complete payload of the MQTT message is checked with the conditions. To access attributes in nested json structures, multiple attribute names can be concatenated by '/', similar to the syntax used for MQTT topics (see the following example):

Examples:

```
Json key:     'foo'
Json data:    {"foo":"bar"}
result:       "bar" will be processed by the Event Manager condition
```

```
Json key:     'foo/bar'
Json data:    {"foo":{"bar":10.27}}
result:       10.27 will be processed by the Event Manager condition
```

Please note, that Json arrays are not supported by the Event Manager!

A condition can be one of:

- Same text
  Content to be checked matches the given text exactly
- Contains text
  The given text is part of the content to be checked
- Value equal
  The content to be checked is assumed to be a numerical value and on successful conversion checked to be equal the configured value
  The event will be triggered after at least once the received value is not equal to the configured value and become equal again with a later message or if the time is over which is configured by the 'Re-trigger event timeout'.
- Value smaller/greater
  The content to be checked is assumed to be a numerical value and on successful conversion checked to be smaller/greater the configured value. If selecting this type of condition, a hysteresis value must be configured. A new event normally will not be triggered on each reception of a MQTT message (which might occur quite often). The trigger shall be executed once the condition matches the first time. For enabling retrigger of the same event, a message containing a value above/below hysteresis value of the condition must be received.

For each of the conditions can be configured a 'Re-trigger event timeout' with preconfigured values between 1 minute and 2 hours. In these cases, a timer will be started on each generation of the configured event type. A new event will only be triggered if this timer has already expired.

Publish mapping Tab

The **Publish mapping** tab allows the configuration of MQTT topics and payloads which shall be added to a publish message during a notification to a MQTT endpoint depending on the event type which has triggered the event plan generating the notification.

In a second configuration step the payload for the publish message must be configured. For a given topic there can be configured several payloads which are selected by the event type and which will trigger the event plan to generate the notification (also to MQTT endpoints). Since no more than exactly one publish message for a dedicated event type is executable, it is not useful to map the same event type and payload with different topics on the same MQTT interface. In those cases, only the first found publish mapping would result in an outgoing publish notification. To avoid those conflicts, it might be useful to configure different endpoints related to more specific topics (see the following example):

Endpoints: tasmota_AF7B08_P1, tasmota_AF7B08_P2 and tasmota_AF7B08_P3
Publish with different publish notifications for POWER1, POWER2 and POWER3 (payload may be 'ON' or 'OFF')

Normally the notification text message generated by the Event Manager is intended to be read by humans and it will not make much sense to use it as payload in a MQTT message in most cases.
If there is a consumer client connected to the MQTT broker which is able and customized to process the

notification text messages generated by the Event Manager (e.g. Node Red) than an MQTT topic can be configured to use the notification text message as payload instead of a payload given by a ‘Publish mapping’. To use the notification text message as payload for the MQTT publish message the flag ‘Message as payload” must have been activated in the ‘Topic’ configuration.

Deletion of MQTT interfaces, topics and endpoints

If MQTT endpoints, topics and interfaces are deleted by the administrator the following rules apply:

- An MQTT interface can only be deleted if no endpoints with assignment to a location are configured for that interface
- On deletion of an MQTT interface all related endpoints, topics, subscribe and publish mappings are deleted implicitly
- On deletion of an MQTT endpoint all related topics, subscribe and publish mappings are deleted implicitly
- On deletion of an MQTT topic all related subscribe and publish mappings are deleted implicitly

Web-API interface

The SIP-DECT Event Manager provides a Web-API that allows other applications, including Mitel CloudLink Workflow, easily to interact with the Event Manager and e.g. trigger events or receive notifications from the Event Manager.

The following event-related actions are supported

- Sending an event to the Event Manager ("reqType":"**event**") and thereby triggering the execution of an event plan
- Canceling the execution of an event plan ("reqType":"**eventcancel**")
- Receiving the result of an executed event plan from the Event Manager ("reqType":"**eventresult**")

The following notification-related actions are supported

- Receive notification from the event manager ("reqType":"**notification**")
- Confirmation of a notification to the event manager ("reqType":"**confirmation**")
- Cancellation of a notification by the event manager ("reqType":"**cancel**"), e.g. if all required confirmations have been received, the event plan has been canceled or there is a timeout

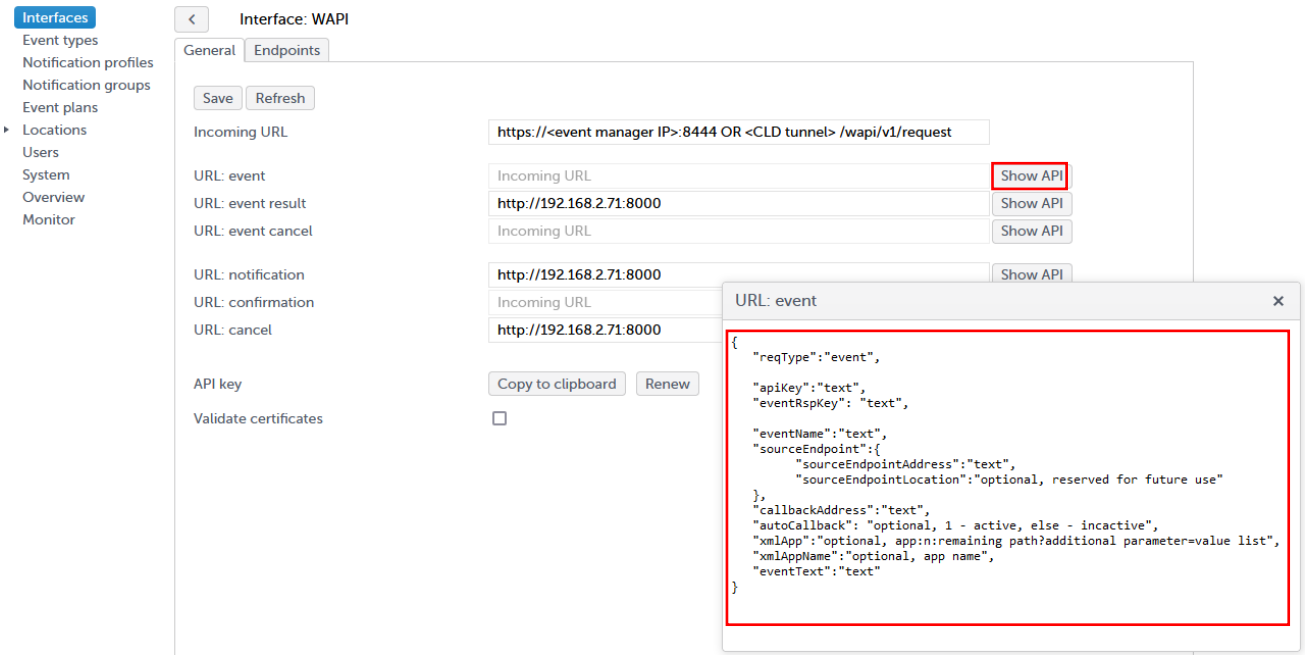
Mitel CloudLink Workflow communicates with the Event Manager via the Mitel CloudLink Daemon, which is integrated into the 4th generation base station. The Mitel CloudLink Daemon is currently not yet available for server installations of the Event Manager, i.e. Workflow cannot reach the Event Manager if it is installed on a Rocky Linux® server for DECT Locating, for example.

The Web-API supports incoming web requests with an URL in one of the following forms:

- `https://<event manager IP>:8444/wapi/v1/request`
- `https://<CLD tunnel>/wapi/v1/request`

The Event Manager accepts http GET and POST requests.

The Json body definition is available via the EM Web GUI by clicking on the “Show API” button for the respective request.



There is also a simplified form for triggering an event in which the mandatory parameters are added to the request as URL parameters and a Json body is not necessary. This means that an event can even be triggered by a web browser, e.g. for testing purposes or by other web applications.

```
https://192.168.2.41:8444/wapi/v1/request?type=event&apiKey=5gDem3N3QS6XcTtViujWwiiO5usOJhDoIQ5NocONjMQMmvwezUEFrIntsTjPFGyz&eventName=SOS&eventText=Test&sourceEndpointAddress=118
```

Example of a request with URL parameters to trigger an event instead of

```
{
	"reqType":"event",
	"apiKey":"5gDem3N3QS6XcTtViujWwiiO5usOJhDoIQ5NocONjMQMmvwezUEFrIntsTjPFGyz",
	"eventName":"SOS",
	"sourceEndpoint":{
		"sourceEndpointAddress":"118"
	},
	"eventText":"Test"
}
```

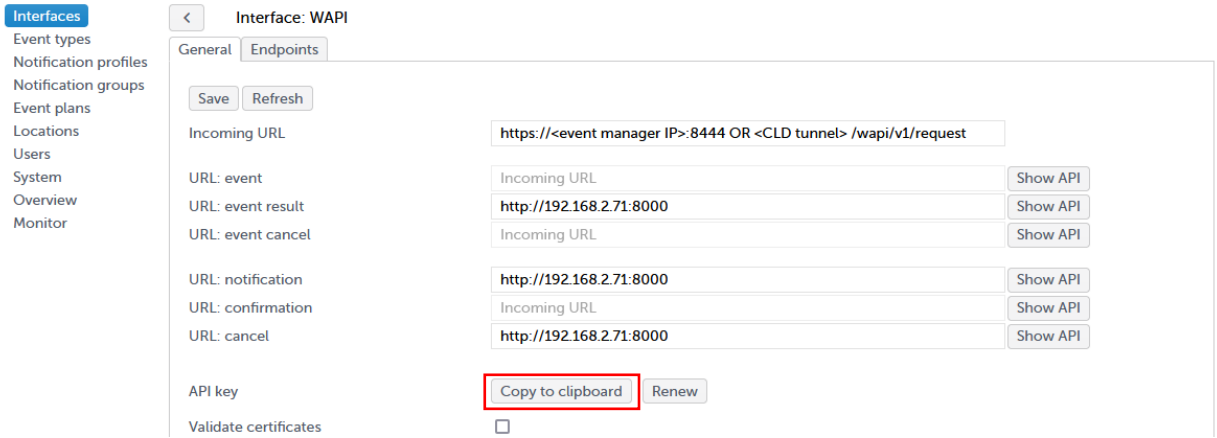
Json body example for the request https://192.168.2.41:8444/wapi/v1/request Content-Type application/json with mandatory parameters only

Starting with release 10.1 the simplified form of event requests via the WAPI interface is enhanced by the additional parameter callbackAddress to support additional use cases with web applications to trigger events via the Event Manager.

The incoming requests (**event, eventcancel, confirmation**) require an API key which can be copied to clipboard by clicking on the “Copy to clipboard” button.

Outgoing request (**eventresult, notification, cancel**) are sent as POST requests with the Json body, whose definition is available via the EM Web GUI by clicking on the related “Show API” button.

The notification Json body contains Event Manager CloudLink Daemon information which are required to satisfy the CloudLink tunnel API which is required to send confirmations back to the Event Manager. They are not relevant for other applications attached to the Event Manager via the Web-API.



```
"eventManager":{
          "eventManagerUUID":"text",
          "eventManagerName":"text",
          "component_id":"text",
          "platform_id":"text"
},
```

The “Validate certificates” options can be used to activate the validation of the certificates of the servers to which the outgoing requests are sent. Further information on handling certificates can be found in the section System / Security Tab.

General Tab

The **General** tab allows configuring the basic settings of the Web-API interface. The following settings can be configured:

- **URL: event result**: URL for outgoing responses to the event requests
- **URL**: **notification:** URL of an external web application (e.g. Workflow) as receiver of notifications from the Event Manager
- **URL**: **cancel:** URL of an external web application as receiver of Event Manager notifications

Examples:

- for Workflow-API:
  https://workflow.eu.dev.api.mitel.io/2017-09-01/webhooks/accounts/ba8750cb-3032-4015-8fde-feddf81da52f/activities/420ed198-5c77-4c14-9117-7330d64b3343/workers
- for WAPI-Tester (a Python application on Windows or Linux to test the Web-API interface;):
  http://10.103.37.79:8000
  The tool can be made available on request for testing purposes without any warranty or support.

The Json body definitions for the requests are available via the respective “Show API” buttons.

The “Copy to clipboard” and “Renew” buttons can be used here to copy or renew the API key that is used for authentication with the web API for incoming requests.

The “Validate certificates” options activates the validation of the certificates of the servers to which the outgoing requests are sent.

Endpoints Tab

The **Endpoints** tab allows the creation of endpoints that can act as event requesters or notification recipients.

IP-Phone Interface

The SIP-DECT Event Manager provides an IP-Phone interface that allows Mitel 6900 SIP phones as well as 6900 MINET phones to connect with the Event Manager via their XML-Application interface to be able to receive event notifications as well as trigger events in the Event Manager. Only one IP-Phone interface can be configured.

To support the connection of a MINET phone to the Event Manager there is the possibility to download an 'Appinfo configuration file' via the IP-Phone interface URIs tab fitting the hardware type of the MINET phone. This configuration file may be imported into the MiVoice Business PBX under "MiVB Web Service Configuration" → "System Administration Tool" → "Users and Devices" → "Advanced Configuration" → "Phone Applications Update". Please note that should a redundant Event Manager be newly configured to repeat these steps, as the Appinfo file changes to accommodate the redundancy.

IMPORTANT: Certificate validation is turned on inside the MINET phone by default and the Event Manager's root certificate needs to be added to the trusted certificates of the phone. Otherwise, the polling from the MINET phone will always fail, which will result in the phone being unable to send or receive event notifications. Alternatively, you may turn off certificate validation altogether if you add the following line to the 69xx.cfg file generated by the Event Manager: https validate certificates: 0

To support the connection of a Mitel SIP phone to the Event Manager, the IP address of the Event Manager has to be added into the phones local configuration e.g. via the Mitel SIP phone's web configurator under the tab "Advanced settings" → "Configuration Server" into the field "XML Push Server List (Approved IP Addresses)" and save the settings. This values in this field are comma separated, so there can be more than one configured IP address (if necessary). In the configuration tab "Advanced settings" → "Action URI" the "Poll" URI (available in the URIs tab of the IP-Phone interface) and a poll interval in seconds must be configured. The recommended smallest interval is 30 (seconds). The "Poll" URL can be copied from the IP-Phone interface tab "URIs". Should you enter multiple Poll addresses, please note that while the intervals between different polling addresses can be different, the larger ones should always be multiples of the smallest one, as per official SIP phone recommendations.

Should a redundant Event Manager exist you must configure its Poll address as well as add its IP address to the "XML Push Server List (Approved IP Addresses)" inside each SIP phone. The redundant Poll URI can be found in the "Poll 2" field and copied from there.

If these configurations have been finished, the phones will poll the Event Manager in the configured poll interval which enables the Event Manager to learn the IP addresses, software-type, hardware-type and language of the phones. It is not necessary to configure the phones with static IP addresses. The phones may be configured with DHCP by the PBX where they are registered.

General Tab

The **General** tab allows the configuration of basic settings of the IP-Phone interface, e.g. the "Validate certificates" attribute for all connected phones. Should certificate validation be turned on, the Event Manager tries to validate the certificates of all incoming connections to the IP-Phone interface i.e. getting polled or receiving an event trigger. The Event Manager does not validate certificates of an IP phone it tries to connect to however, such as sending a notification to a phone. It is solely for incoming connections.

Endpoints Tab

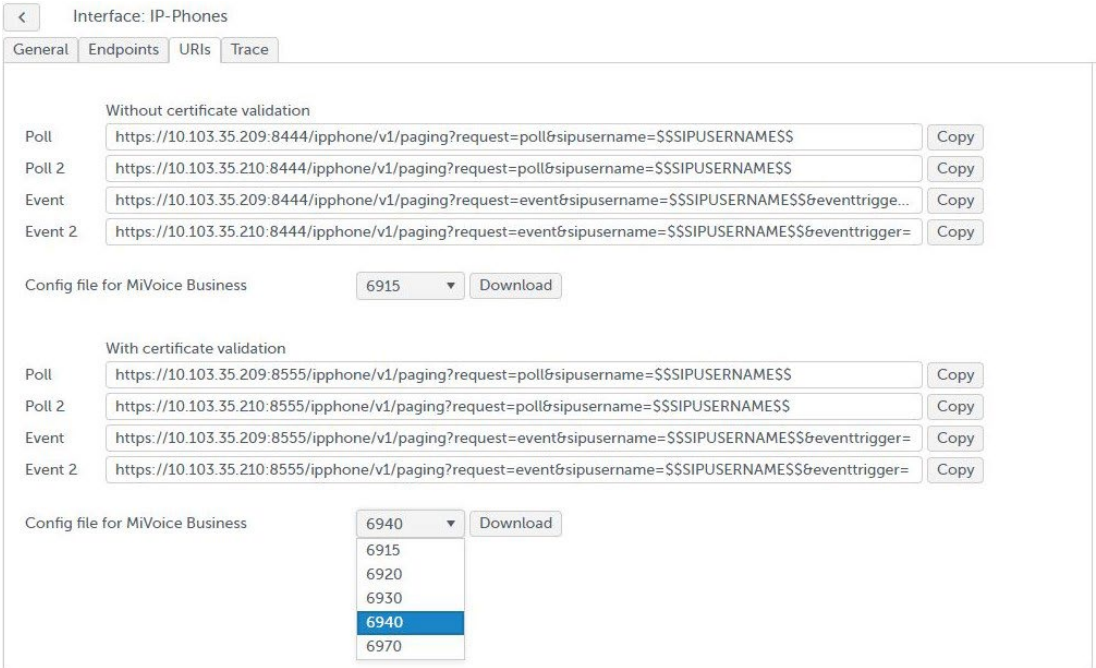
The **Endpoints** tab allows the creation of endpoints that can act as notification recipients or event requesters.

By configuring endpoints, the Event Manager is able to accept polling requests from the phones and will complete the endpoints internal records by the IP address as well as various other needed data to be able to send notifications to that IP phone. Once a valid poll has been received by the Event Manager from a configured endpoint, its record will be updated and show its current IP address. Clicking on it directs you to that phone's web configurator if you are in the same network.

In order for polling requests as well as event requests to be accepted by the Event Manager from an IP phone, the endpoint's "Address" field needs to match the phone's SIP username. Also a "Label" (e.g. a username) and a "Location" can be assigned to an endpoint record.

URIs Tab

The **URIs** tab allows you to download configuration files for different types of MINET phones and to copy URIs, e.g. for polling the Event Manager (Poll, Poll 2) or for triggering events by Mitel SIP phones (Event, Event 2).



The Poll URIs are the URIs the phone uses to tell the Event Manager as well as the redundant Event Manager that it is reachable and what IP address and device type it has. The Event URI is used to configure a key on an IP phone which can be pressed in order to trigger an event inside the Event manager. Both Poll and Event URIs are available as "Poll" & "Poll 2" or "Event" & "Event 2" respectively. The fields with 2 in their

names contain the URIs that lead to the redundant Event Manager and should be utilized to configure an IP phone to be able to communicate to both the active and inactive Event Manager in case of a failover.

The URIs as well as the configuration file are available as two different types, one without certificate validation (connections from the phones will be received on port 8444) and one with certificate validation (connections from the phones will only be received on port 8555).

Should an IP phone be configured to use the addresses supplied under “With certificate validation”, the certificates of the phone will always be validated, even if “certificate validation” under the interface’s “General” tab is turned off. If the validation fails, the request is discarded and will not be processed.

Should an IP phone try to trigger an event or poll the Event Manager with an address under “Without certificate validation” while the interface’s certificate validation is turned on, the request will be discarded and not processed.

	Certificate validation on	Certificate validation off
Port 8444 – no cert. val.	no	yes
Port 8555 – cert. val. Success	yes	yes
Port 8555 – cert. val. Failure	no	no

Triggering events from an IP phone

In order to trigger an event from an IP phone, a specially configured softkey is required. For SIP phones, enter the phone’s WebUI under “Softkey and XML”, configure a Softkey with the type XML and give it a fitting name. For MINET phones, enter the System Administration Tool, go to Users and Devices, enter the User and Services Configuration, select the phone you wish to give the XML softkey to and select the tab “Keys” and configure an “URL Line” key.

Now for both phone types, paste the Event URI into the “Value” or “URL” field and add the full name of event type that this phone can trigger behind the keyword “eventtrigger=” in the URL. Event type names can be found under “Event types” in the table column “Label”. They are case sensitive. There is also the possibility to add the URI parameter “callback=” together with a valid callback number and the URI parameter “eventtext=” with a custom event text. Don’t forget to add the “&” character between these URI fields! A fully configured URI could look something like this:

```
https://x.x.x.x:8444/ipphone/v1/paging?request=event&sipusername=$$SIPUSERNAME$$&
eventtrigger=SOS-Key&eventtext=my own SOS event text&callback=1234
```

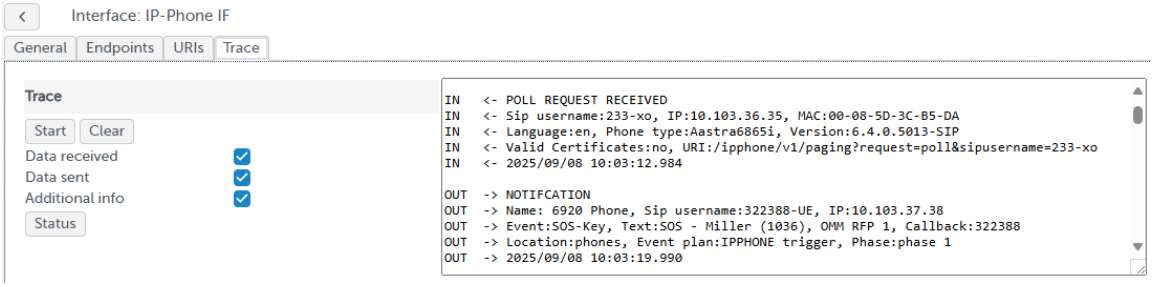
After pressing the newly configured button, the event will be triggered in the location of this endpoint. If an event was successfully triggered, a positive confirmation is displayed on the IP phone’s screen in the form of either a green checkmark or with the text “ok” depending on the phone’s capabilities. This is only possible if the event type is configured inside the event manager, an active IP-Phone interface exists in which the event triggering phone is correctly configured and set to active and the location in which this IP-phone is configured has an active and currently working event plan that can process this event. The feedback will be a negative one instead if any of these criteria are not met and no event could be triggered. The negative feedback will either be a red X or the text “X” depending on the phone’s capabilities.

Should a redundant Event Manager be configured, two softkeys must be configured for each event trigger with the only difference being the IP address. The first button triggers an event in the first Event Manager and the second button triggers an event in the second Event Manager. Depending on which Event Manager is the

active one, the corresponding button needs to be pressed.

Trace Tab

The Trace tab allows a system administrator to see what data is getting send and received on that particular IP-Phone interface as well as additional information and error messages that could occur during processing.



You may select what data you want to see output in the trace window. Error messages will always be displayed. In order to start outputting trace information, press “Start”. You may also stop the trace afterwards or clear the trace window of all information.

While the trace has been started, you may request an endpoint status to be output in the trace window via clicking the button “Status”. After pressing it, the IP-Phone interface will compile a comprehensive report on all configured endpoints. First it will list general stats (how many endpoints are configured, how many endpoints have polled recently, how many endpoints are reachable via their currently known IP address) and then it will list all known problems with every endpoint. This can be used to verify if all endpoints are reachable. A notification will be sent to all endpoints that will display that such a status report was requested manually.

Limits

The Event Manager allows up to 100 IP phone endpoints to be configured on the RFP4G platform and up to 1000 IP phone endpoints to be configured on a server installation.

The SIP phone’s software version should be 6.4.0.5013 or higher.

The MINET phone’s software version should be 03.00.00.052 or higher.

The 6905 and 6910 phones cannot dial a supplied callback number.

MINET phones currently cannot ring when receiving an event as their XML-application interface lacks the capability to do so.

Currently, the Event Manager may only take a maximum of 9 seconds per notification sending for a phone. Should establishing a connection and sending the XML-content take longer than 9 seconds, the sending process will be stopped and the notification will be discarded.

The Event Manager on an RFP4G can only send 100 notifications to IP phones at the same time. Should this limit be reached, the next 100 notifications will be queued and wait until the actively being sent messages have been reduced to under 100 by either timing out or being successfully concluded. In total, 200 notifications may exist at a time inside the Event Manager. All notifications to be sent that exceed that amount will be discarded without being retried. Furthermore, only 200 concurrent cancel or direct answer messages can be sent at a time.

The Event Manager on a server installation can instead send 2000 notifications to IP phones simultaneously, queue up to 1000 more notifications and send up to 3000 cancel or direct answer messages at a time.

Web event Interface

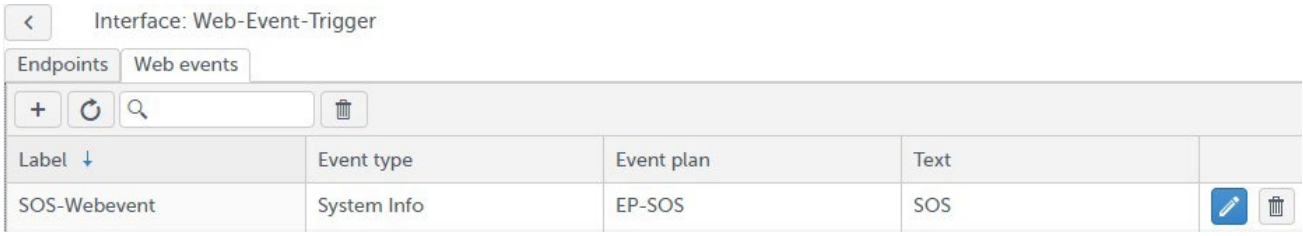
The SIP-DECT Event manager provides an interface for the triggering of events by a logged in user directly from the web application. There are two tabs available in the Web event interface configuration for adding web user endpoints and for configuring web events that should be available for all authorized users to trigger.

Endpoints Tab

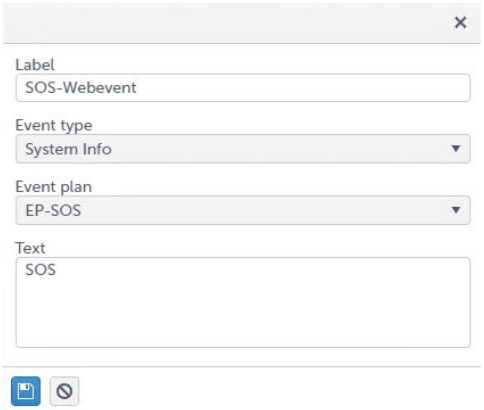
The **Endpoints** tab allows to create endpoints for all those web users that should be allowed to trigger web events. The here configured endpoints will be assigned automatically to the root environment to allow them to execute of an arbitrary event plan. The creation of endpoints in this interface is also necessary for the feature 'Locating alert' as part of the Locating application. If a web user will be deleted from the Users pane, the assigned web event endpoints will also automatically be deleted.

Web events Tab

The **Web events** tab allows to define web events that should be available for authorized web users to trigger directly from within the Monitor pane or from the Monitor tab in the Locating pane.



During the creation of such a web event an event type and an event plan have to be selected from dropdown lists and a predefined event text can be configured (may be modified later during the event triggering). All available event plans are selectable independently from there assigned environments or schedules. They will be triggered and executed later also independently from the environment and without observing schedules.



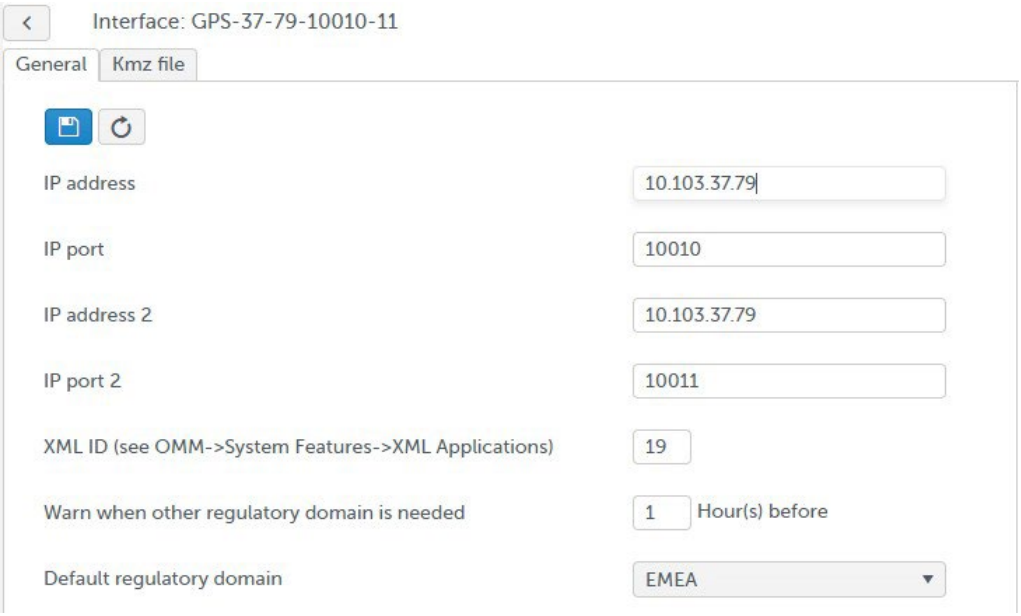
GPS interface

The GPS interface is general available but only suitable for Cruise line business to detect the need of switching the DECT regulatory domain of the SIP-DECT system based on the position, the speed and the direction of movement of the cruise line ship.

General Tab

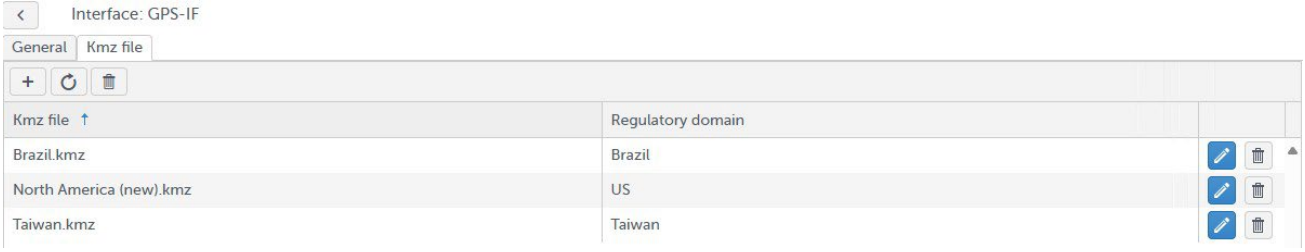
The **General** tab allows configuring the basic settings of the GPS interface. The following settings can be configured:

- **IP address 1**: IP address of first GPS data server
- **IP port 1**: IP port of first GPS data server
- **IP address 2**: IP address of second GPS data server
- **IP port 2**: IP port of second GPS data server
- **XML ID** Same value as configured in SIP-DECT OMM/OMP for EM-XML-Menu
- **Warn when other regulatory domain is needed:** Hour(s) before the need to switch
- **Default regulatory domain**: Setting of the default DECT regulatory domain (no KMZ file needed)



Kmz file Tab

The **Kmz file** tab offers the availability to upload the KMZ data files with the geographical data of the polygons that describe those areas that are defined for the special DECT regularity domains. The files may be received from Mitel support. Please note, that uploaded KMZ files are not part of the EM database backups!



The GPS interface connects the Event Manager with up to two GPS data servers which deliver NMEA data

records that containing the following data:

- actual position
- actual speed
- actual direction of movement

The interface supports GPRMC and GNRMC data records.

The state of the interface is considered as ok (green point in the web admin's interfaces tab) as long as at least the connection to one GPS data server is working and this server is delivering data in the correct format with correct checksum. But if there is configured a second data server and this server could not be connected, the GPS interface will raise one time an event of type 'System info' with a data error (link could not be connected). This case will also lead to display the state of the interface as misconfigured (yellow point in the interface tab of web admin) until the misconfiguration has been solved by correcting the link data or by deleting the link in the interface configuration.

The GPS interface connects and reconnects to both data servers (if configured), but the received data will only be processed from one server (called 'active link').

The interface will switch automatically to process the data from the second data server (called 'standby link') in case non or faulty data have been received from the active link for more than three minutes or in case the connection to this link got broken (link is disconnected). In this case the state of the interface will switch to an error state (red point in the interface tab of web admin) as long as there are no data received from the formerly standby and now active link.

In case of receiving non or faulty data from both data servers or in case of a broken connection the interface will raise an event with event type 'System info' to generate notifications (e.g. on DECT phones or via SNMP traps) and it will also try to reestablish the broken connection.

The Event Manager read the configured DECT regulatory domain from the OMM configuration at every connect or reconnect of the OMM AXI connection and it will also be informed via this connection if the configuration in OMM has changed.

The Event Manager can change the DECT regulatory domain via OMM AXI.

The Event Manager's web service offers for authorized DECT user a XML application which allows those users to change the DECT regulatory domain on the OMM.

The XML application includes two GPS related menus / actions:

- Select and activate any available DECT regulatory domain in the OMM configuration.
  This application is not offered via a link inside an Event Manager notification.
- Confirm the change of the DECT regulatory domain predicted by the Event Manager GPS interface.
  This application is offered via a link inside an Event Manager notification to the DECT handset of authorized DECT users.

The setup of the GPS XML application(s) has to be done manually in OMM configuration (via web admin or OMP) with the following setting under System features / XML applications:

EM-Menu https://EMAddr:8444/evmMenu/?ppn={ppn}&uid={uid}&sipusername={number}

Only a few SIP-DECT users shall have the right to change the DECT regulatory domain of the SIP-DECT system. As there is no way to offer the XML application only to dedicated SIP-DECT users, the Event

Manager restricts the execution of this application to only those users which are configured in the Event Manager configuration by adding them to the fixed available notification group ‘GPS’ (which will be automatically created if a GPS interface is configured in the Event Manager). Only SIP-DECT endpoints with terminals of type 700d are supported to be members of the GPS notification group (due to special capabilities for message handling).

Other ways of changing the DECT regulatory domain of the SIP-DECT system (e.g. via OMM web service or OMP) can be used independently from the Event Manager mechanisms.

Processing GPS data depending on Event Manager configuration

The Event Manager will generate two different events depending on the configuration and based on current position:

- Warning (event ‘GPS Warning’)
  will be generated at the configured time before the need for switching the DECT regulatory domain is reached (based on the current position, actual speed and actual direction of movement).
- Error (event ‘RegDomain Err’)
  will be generated in case the actual position a different DECT regulatory domain is needed than actually configured in SIP-DECT

The warning event is only fired once, but

- can occur again after changing the DECT regularity domain
- can occur again if the DECT regularity domain of predicted position changes or is again outside any configured regularity domain polygons in the KMZ files

The DECT regularity domain will never be automatically changed by the Event Manager. The switching must be executed manually by an authorized SIP-DECT user via the XML application provided by the Event Manager in the notification or via other configuration tools (like OMM web admin or OMP).

If for an actual position another DECT regularity domain than currently configured is needed, an event “DECT Reg Domain Error” for the root environment will be fired by the GPS interface. An event plan must be available to handle this and notify the SIP-DECT user members of the notification group ‘GPS’. The group exists by default, the members must have been added to this group before.

The event “DECT Reg Domain Error” is fired once per hour until the configured and needed regularity domain matches again.

The SIP-DECT users who receiving this notification will get a link to an XML application within the notification that directly allows to change the DECT regularity domain from within this application to the required domain.

There is no algorithm that introduces a hysteresis in case the ship travels along the border of a polygon. Changes of the predicted DECT regularity domain just reset the warning state machine, the ‘GPS Warning’ event will be fired again if the predicted regularity domain differs from the actual domain.

To avoid notifying change requests for regularity domain due to a single GPS data record (which may have

some inaccuracy) there must be received GPS records in sequence pointing to a changed regularity domain polygon before a ‘GPS Warning’ event or ‘DECT Reg Domain Error’ event is raised (normally one GPS data record will arrive per second).

Each time a change of the DECT regularity domain is detected this event will be logged into the Event Manager’s logfiles with the information which domain is active now.

Each time a change of the DECT regularity domain is requested via the XML application by a SIP-DECT user this will be logged with the information about the requesting user and the requested new domain.

If there is configured other domain than ‘None’ in the “Default regularity domain” on the GPS interface General tab, this will be the valid DECT regularity domain for all positions which not belong to any of the configured polygons described by the loaded KMZ files. This implies that a ‘DECT Reg Domain Error” will be fired every hour if the ship has left a region defined by a KMZ file until the DECT regularity domain is changed to this “Default regularity domain”.

Event types

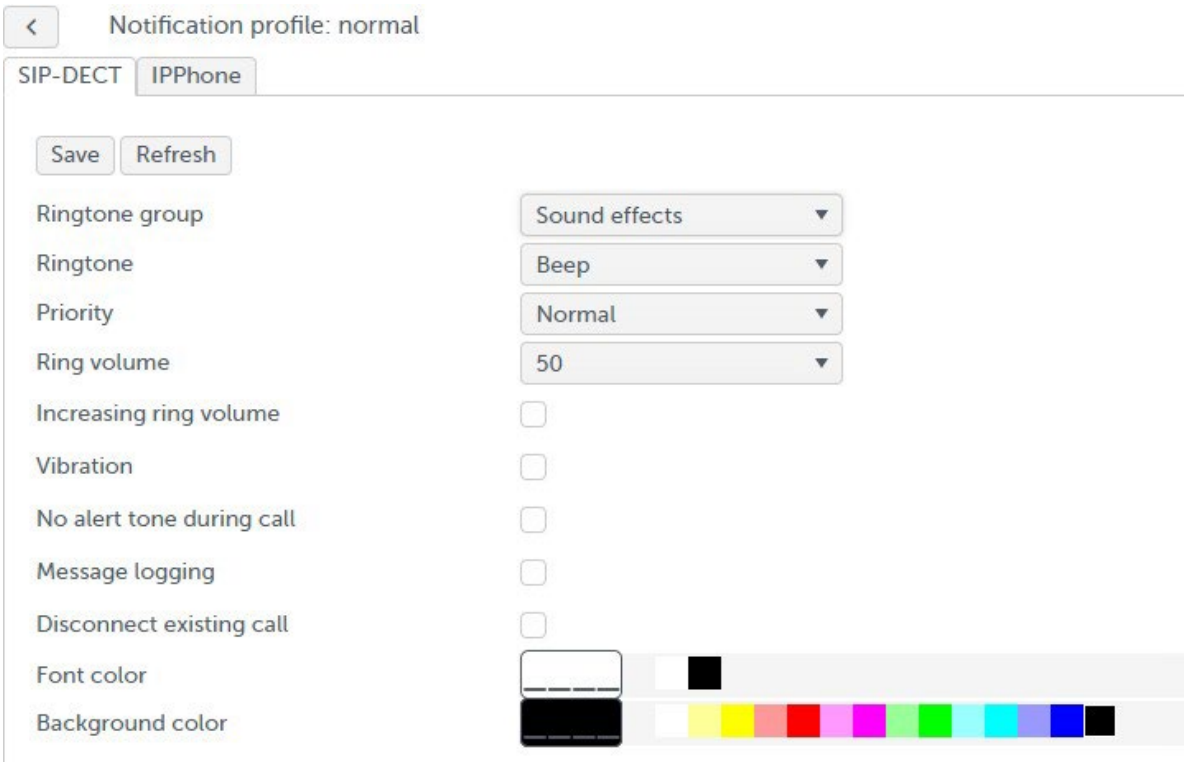
There are eight default Event types (‘Escape’, ‘Man Down’, ‘No Movement’, ‘SOS-Key', ‘System Info’, ‘Locating Alert’, ‘GPS Warning’, and ‘RegDomain Err’) available. These types can be changed but cannot be deleted. The default Event types ‘Man Down’, ‘No Move’, ‘Escape’ and ‘SOS-Key’ correspond to the Alarm triggers which are also available as standard in SIP-DECT. The Event type ‘System Info’ is automatically generated in the location ‘root’ by all interfaces when they change their status. This occurs, for example, when an ESPA-Interface loses its connection to the configured IP-address or when connection to the OMM is lost in the SIP-DECT-interface. Therefore, it is useful to make an event plan in the location ‘root’ for the event type ‘System Info’ in order to catch all important changes in the event manager’s functionality quickly.
To handle additional Alarm triggers that may be defined in SIP-DECT OMP, Event types with the same name or short name as the name of the Trigger ID in OMP must be configured in the SIP-DECT Event Manager.
All Event types serve as a kind of filter in an Event plan to control the escalation of an event. Based on the assigned priority the system knows in which order the events should be processed. Important events should therefore be configured with a higher priority.

Note: An event displayed on a SIP-DECT terminal will be overwritten by a higher priority event.

Notification profiles

A Notification profile determine how a notification should be presented to recipients (SIP-DECT or IP-Phone). It is assigned to the receiving endpoint or notification group within event phases. Only one notification and only that one with the highest priority (Event type priority) is displayed on a phone. Notifications with lower priority are not transmitted to the phone if a message with higher priority is to be displayed. If there are several messages with the same priority at the same time, they will be transmitted one after the other to the phone, with each message being displayed for at least 20 seconds before it is replaced by the next one (only valid for SIP-DECT phones). One notification profile (‘normal’) is available by default, this profile cannot be deleted. Click the link under the column ‘Label’ to change the profile settings (Melody, Ringtone, Volume, etc.) for a profile. From version 10.1, the notification profile settings are divided into those for SIP-DECT phones and those for IP phones in different tabs.

Notification profile settings for SIP-DECT



A Ringtone group is a set or collection of ringtones that can be assigned to specific contacts, groups, or categories. Ringtone groups are used to customize the incoming call alert sounds for different callers or types of calls. The ringtone group can be specifically selected from all the ringtones available from SIP-DECT.

If the ‘Increasing ring volume’ option is used, the ringtone starts quietly and then gradually reaches the ring volume set. In addition, notification can also be signaled by telephone vibration (if supported by the phone type).

If the ‘No alert tone during call’ option is active, a notification is delivered without acoustic signaling while the terminal is on a call. If ‘Disconnect existing call’ is selected, an existing call will be disconnected at the time of the notification.

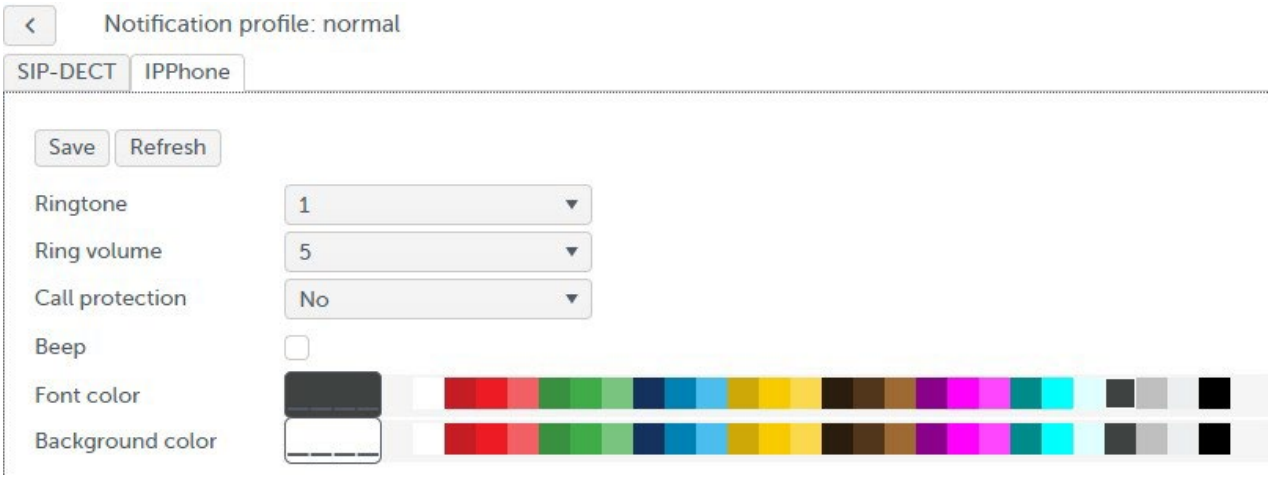
If ‘Message logging’ flag is enabled, answered notifications (accepted or rejected) will remain available in the text messages list on the Mitel DECT phone for up to fifteen messages. Further messages will overwrite the oldest message entries in the list. Not answered messages (neither accepted nor rejected) will not be logged in the text messages list on the Mitel DECT phone.

If the telephone supports ‘Font color’ and ‘Background color’, the font and color display of the message can be controlled by the SIP-DECT Event Manager.

Restrictions and behavior:

- Settings not supported by the used telephone are ignored.
- ‘Priority Low’: ‘Ringtone group’, ‘Ringtone’, ‘Ring volume’ and ‘Increasing ring volume’ has no effect.
- ‘Priority Emergency’: Pop-up window during call only available with this priority
- Further information about the behavior of displayed messages: Please see the document ‘Mitel 600/700 DECT Phone Messaging and Alerting Applications’!

Notification profile settings for IP Phones



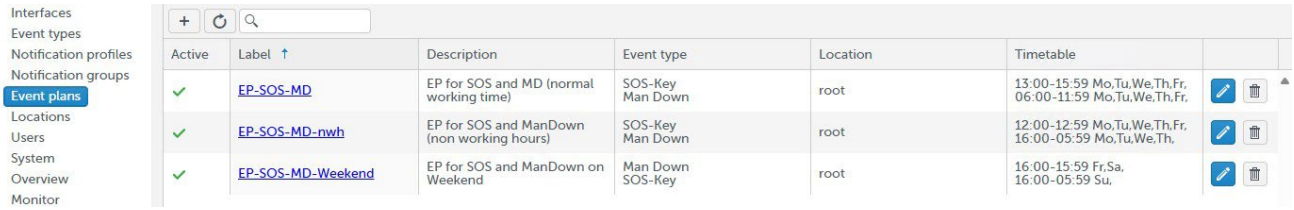
For IP phones less settings are available than for SIP-DECT phones because of the limited possibilities of the IP-Phones. Ringtone off or Alarm 1..7, ring volume 1..10 and a mix of font color and background color is configurable as well as a flag for Beep (important especially for Minet phones) and “Call protection”.

Notification groups

Endpoints that can receive notifications can be combined into a notification group. This simplifies the configuration regarding the escalation of an event. If the assigned notification group address matches with the source endpoint address then the “Use call address" feature of the event phase can be used.

Event plans

Event plans describe how to react to certain types of events that occur at different locations. Event plans can consist of up to 10 escalation phases and define the process for handling these events and the resulting notifications in the different phases.

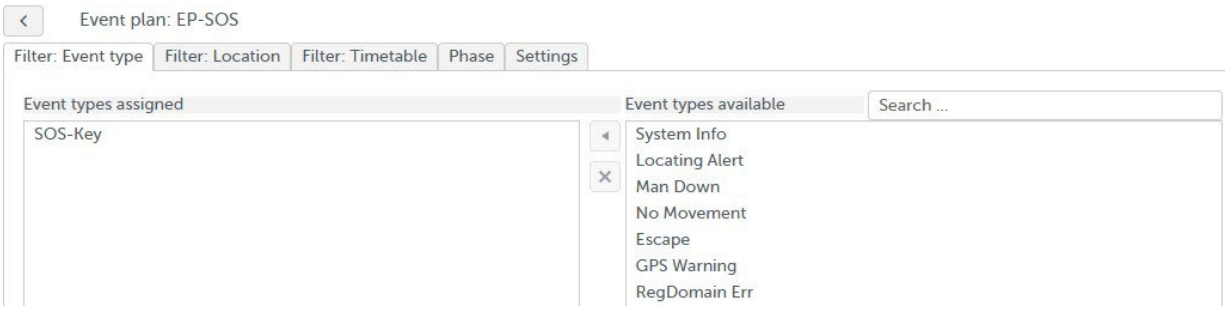


The ‘Event plans’ pane shows an overview of all configured event plans with the event types they will handle and the locations they will be executed from, and additionally since release SIP-DECT 10.1 the configured timetables the plans are valid.

The following settings can be carried out in the **Event plans** configuration pane:

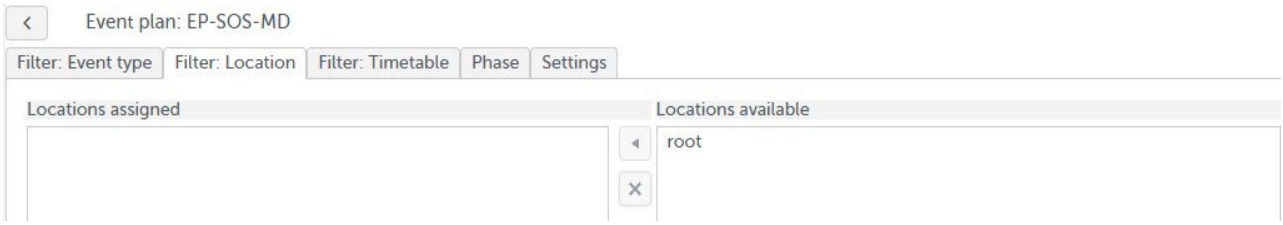
Filter: Event type Tab

Different types of events can be assigned here to the Event plan. At least the following default Event types are available: **System Info**, **SOS-Key, Man Down, No Movement, Escape, GPS Warning, Reg Domain Err, and Locating Alert.**



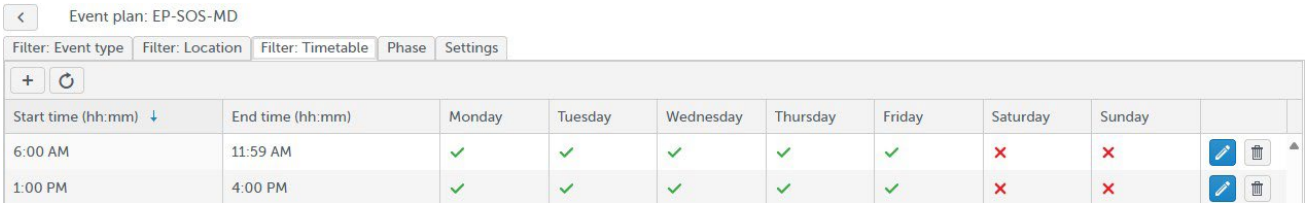
Filter: Location Tab

Formerly created locations (to which endpoints are assigned) can be assigned here to the Event plan.

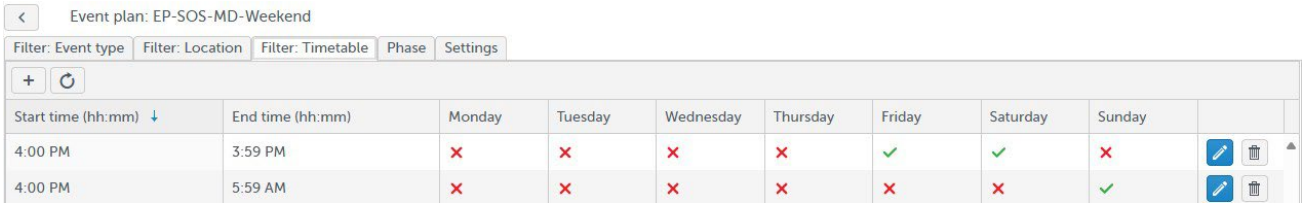


Filter: Timetable Tab

Schedules can be assigned here as a filter to the Event plan. The following schedule is an example for normal working hours between 6:00 am and 4:00 pm with a break between 12:00 pm and 1:00 pm.

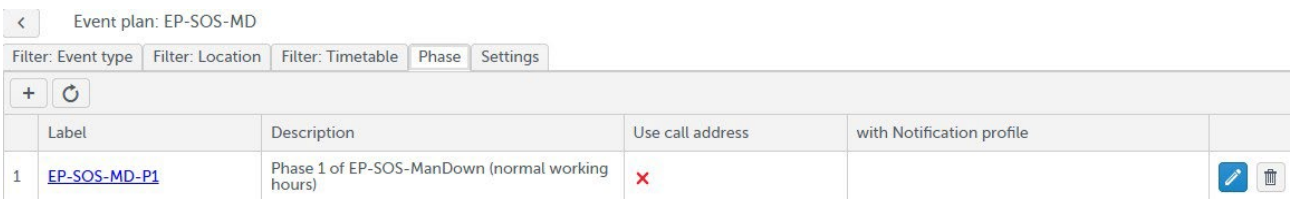


The configured start time is assigned to the selected dates of the week, the end time is associated to the following day if it is set to a value later than 11:59 pm. The following example shows a weekend plan from Friday 4:00 pm to Monday 6:00 am.

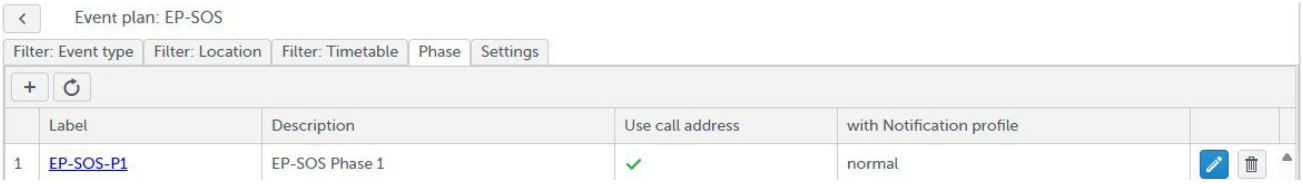


Event plan Phase tab

In the **Event plan Phase tab** can be configured up to 10 phases for escalation with different destinations, different timings, and different settings regarding necessary confirmations.



By editing the phase settings, the ‘Use call address’ flag can be enabled, and a notification profile may be assigned. With this kind of configuration, a direct assignment of call addresses to a notification group with this address can be realized. In the incoming interface (e.g. ESPA) an endpoint with this call address must be configured.

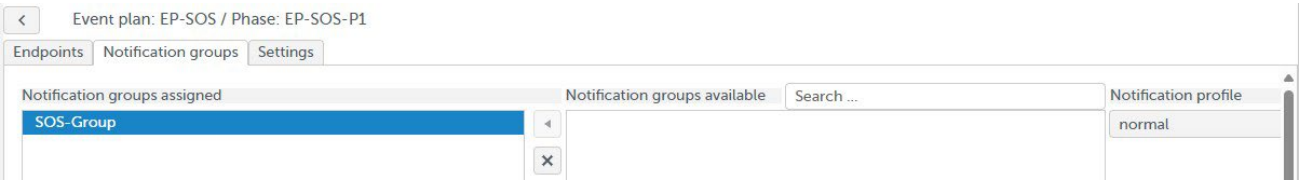


Phase Endpoints Tab

Up to 1000 endpoints can be added to a phase or deleted from a phase in the Phase Endpoints tab. To each endpoint a formerly created notification profile can be assigned here also.

Phase Notification groups Tab

Up to 50 notification groups can be added to a phase or deleted from a phase in the Phase Notification groups tab. To each notification group a formerly created notification profile can be assigned here also.



Phase Settings Tab

The following settings can be carried out in the **Settings** tab for a phase:

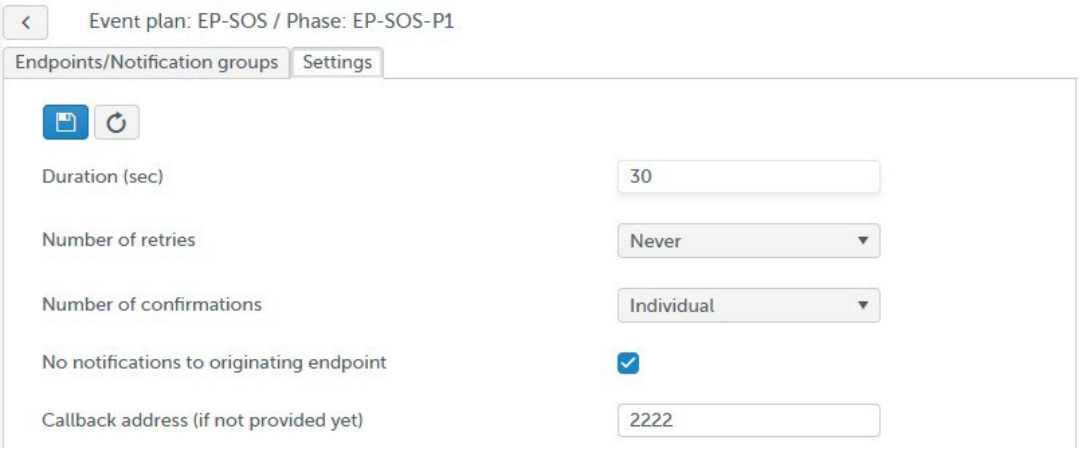
- The duration in seconds for this phase
- Number of retries (repetitions of this phase)
- Number of confirmations (needed for successful ending of the phase)

Note: ‘Individual’ implies that in this phase all assigned endpoints must confirm the received notification before the phase ends successfully. If the number of confirmations is not reached, it moves on to the next phase (if configured), is repeated (if configured) or is terminated with a timeout result after the phase has expired.

Note: If there are assigned outgoing endpoints like Modbus or SNMP to a phase, the setting for the number of confirmations should not be set to ‘Individual’ to avoid unsuccessful phases (because those types of endpoints will never be able to confirm received messages).

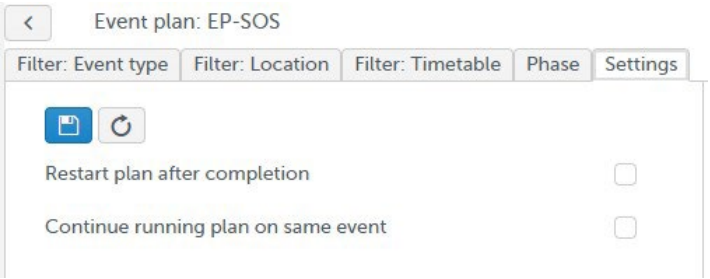
From version 10.1, there is an additional flag in the phase settings configuration available to avoid notifications to the originating endpoint.

By adding a callback address additional use cases can be configured (if this attribute is not included in the event request by any endpoint). This callback address would then be included into the notifications to the destinations of this event phase. DECT handsets which would receive those notifications would so be able to direct dial this callback address when pressing the green hook off key.



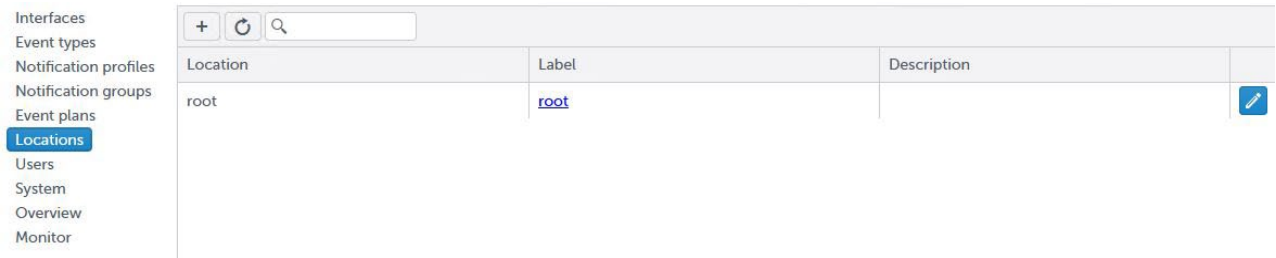
Event plan Settings Tab

Via the **Settings** tab of an Event plan can be configured general settings for this Event plan. A running event plan is terminated and restarted if the same event is sent from the same endpoint. This can prevent the execution of further phases. Since SIP-DECT 10.0 the option is introduced that a running event plan continues to run and further events of the same type from the same endpoint are ignored until the running plan is terminated.



Locations

By defining locations, spatial environments can be mapped into a tree structure. A location means the origin of an event.



The root location is always present and cannot be deleted.

To create a new location, a location in the Locations tree must be selected, and the button + must be

pressed. The new location is then based on the location that was selected before. The behavior has changed in release 10.1, selecting lines in the location tab is no more needed to create new locations.

All endpoints that should be used to trigger events can be assigned to a desired location by following the link under the column ‘Label’. The assignment can also be changed via the **Endpoints** tab in the **Interfaces** configuration pane. Endpoints that are not assigned to a any location cannot trigger an event.

User

The **User** pane allows to create, edit and delete users and to change the passwords of the users. The default user ‘admin’ with permission ‘Configuration’ cannot be deleted. There are two other user permissions available for ‘Monitoring’ and ‘Locating’ which can be used to add users with those different permissions.

System

The **System** pane consist of the following tabs:

General Tab

The following settings can be carried out in the **General** tab of the system:

- A system name which subsequently will be displayed also in the headline of the Event manager web application.
- CloudLink daemon can be enabled here (for remote management of the Event Manager)
- CloudLink status is shown here (running or not running)
- The version of the running Event manager application is shown here
- Redundancy configured (IP addresses of configured redundancy EM instances)
- Redundancy connected
- An external IP-Watchdog outside of the system can be configured here which observes a ping from the Event Manager (normally sent at regular interval every 30 seconds as long as it is working correctly). The IP-Watchdog can trigger an alert by Email, SMS or SNMP Trap, or activate a relay for interruption of power for the monitored device to restart the RFP where the Event Manager is configured in case of missing ping from the monitored device.

Backup/Restart Tab

The following actions can be carried out in the **Backup/Restart** tab of the system:

- **Restart**: The SIP-DECT Event Manager can be re-started with this menu item. The SIP-DECT Event Manager is briefly unavailable.
- **Restart with factory defaults**: All data and settings on the SIP-DECT Event Manager are irreversibly deleted when the factory defaults are restored during this type of Restart.
- **Export log**: Log files will be downloaded from the SIP-DECT Event Manager. The log files consist of two csv files which contain the event summary and the event execution details. Depending on the traffic on the Event Manager there are saved the logs from the last days or weeks (maximum size of the details log is 6 MByte).
- **Export config**: A running configuration of the SIP-DECT Event Manager will be downloaded and saved on the local computer of the administrator.
- **Import config**: Allows to restore an existing configuration to the SIP-DECT Event Manager as zipped file (extension “.gz”) but also as normal text file. A validity check is conducted before activation. A configuration recognized as defective or incomplete will not be activated. During the import of an existing configuration the user data will be used from the running SIP-DECT Event Manager system.

If the configuration file was recognized as complete the SIP-DECT Event Manager system will be restarted automatically to activate the imported data backup.

Security Tab

The following actions can be carried out in the **Security** tab of the system:

- The import of additional trusted certificates which are needed to validate certificates used in the SIP-DECT OMM (for future use) or for interfaces like MQTT or Web-API.
- The import of a local certificate chain and private key (with or without a password) for the SIP-DECT Event Manager which will then be used for the web access to the Event Manager application.
- Via a 'Delete' button formerly installed certificates and private keys can be deleted at once.
- Via a dedicated 'Restart' button the activation of newly imported certificates or private key into the system will be finalized (import into web server configuration).

If a trusted certificate or a local chain certificate has been installed the number of those certificates will be displayed. There is also visible if a private key has been imported. The names of files with trusted certificate(s) are also displayed in a separate table on this page. Trusted certificates can be deleted again from within this extra table. The local certificate chain and private key can only be deleted again together.

If a local certificate chain was imported, the corresponding private key (and configuration of needed password) must strongly be done also before a restart of the SIP-DECT Event Manager. Otherwise the system will possibly be unreachable for further configuration via the web admin.

Security level Tab

The following actions can be carried out in the **Security level** tab of the system:

- Setting of a security level for the Event manager application (High, Medium, Legacy)
- Configuration of 'Used Cipher Suites' for the different Security levels

Normally there is configured as a default the security level 'High' and a default setting for 'Used Cipher Suites'. These settings may be modified here carefully. Therefor a list of the currently configured and of the general configurable cipher suites is shown here. The addition of cipher suites into the 'Used cipher suites' could be managed by selecting the cipher suites name from the table entry 'Supported cipher suites' with a semicolon in front of it at the end of the listed cipher suites in the upper list entry (Used cipher suites). An entry can simply be deleted from the 'Used cipher suites' by editing the table entry after deselection of the 'Use defaults' checkbox. In all cases of changing Cipher Suites, the configuration must be finished by pressing the 'Save' button.

Console Tab

The **Console** tab offers access to the Event Manager console via the EM#s Web GUI for support purposes. This eliminates the need of SSH to establish an additional SSH connection via a corresponding client. This is particularly important when accessing the Event Manager remotely via the Mitel Cloud. The access is limited to authorized users with the configuration profile (admin) and access to the Linux shell is omitted. Tthe console can be used exclusively at the same time either from the Linux shell (SSH terminal) or from the web service. The secure remote access via the Mitel Cloud is only possible if the Event Manager is running on a RFP4G base station.

CloudLink Tab

The **CloudLink** tab is only visible if the CloudLink daemon has been enabled before in the General tab (only

possible if the Event Manager is running on RFP4G base station). Via this tab a detailed CloudLink Daemon window will be available to connect the Event Manager with the CloudLink portal and to start the tunnel for the remote access to the Event Manager via the Mitel Cloud.

Information about the CloudLink Daemon portal and system inventory in the CloudLink Portal will be available with the CloudLink documentation on the Document Center at

https://www.mitel.com/document-center/technology/cloudlink.

An account with SIP-DECT integration is needed on the CloudLink portal.

Before removing the OMM or Event Manager from an RFP, the tunnels must be stopped and the CloudLink Daemon must be disconnected from CloudLink.

The CloudLink Daemon connects to *.mitel.io services via https (port 443).

Overview

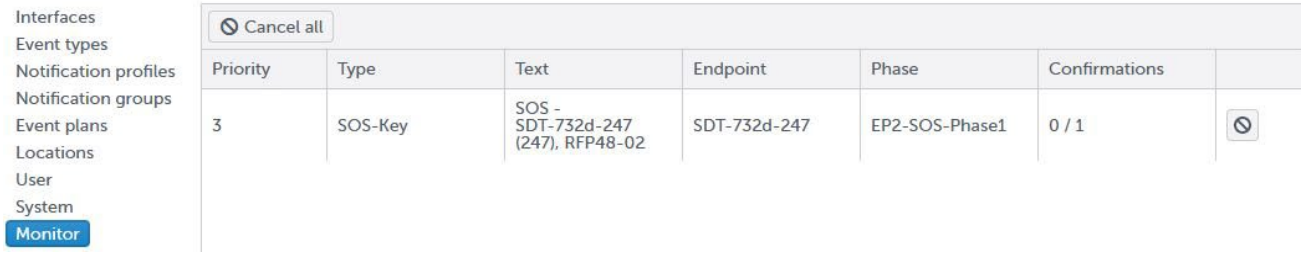
The Overview pane, redesigned in release 10.1, can be used with different views, some of them with specific filters, to provide a more detailed and manageable representation of relevant configuration data.

The following display filters are available to get better overview experience:

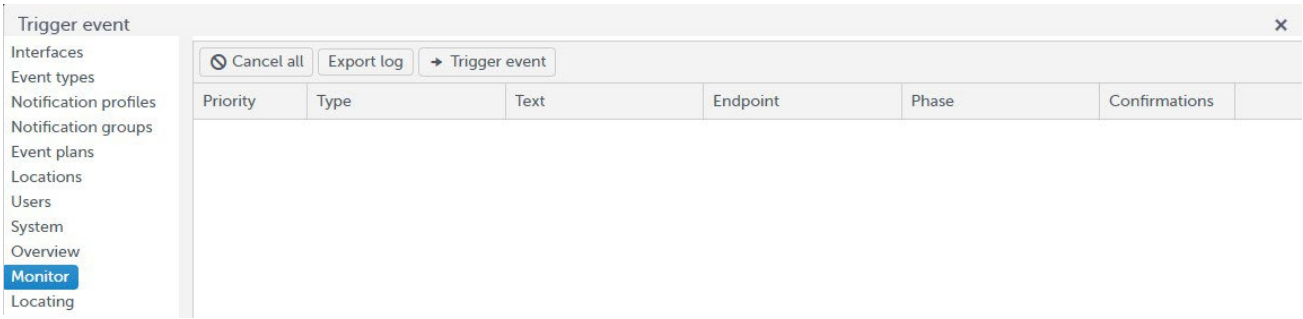
- Event flow
- Plan execution flow
- Notification groups
- Interfaces endpoint relations
- MQTT mappings

Monitor

The **Monitor** pane shows a table with the currently active event handlings. Single event lines from this table or all active event handlings can be canceled from this point.



If a Web event interface is configured, in the Monitor pane there is also available a “Trigger event” button at the top of the table which may be used to select predefined trigger events from a list to execute the related event plan. Via the “Export log” button the EM logs (summary and details) can also be downloaded here.



Event Log (Summary and Details)

The event logs summary and details can be downloaded via the web admin as .csv files
As of SIP-DECT 10.0, the information has been improved so that there is now a clear indication that a notification has been received by the DECT phone.

Column	Information	Meaning
Status	Notify	Notification was sent to the DECT phone
	Notification received	Notification was received by the DECT phone
	Confirmed	User has confirmed the message (positive or negative)
	Notification terminated	Notification was terminated by the EM
Confirmation	Accepted	User has positively confirmed the notification
	Rejected	User has negatively confirmed the notification
	Not confirmed	User has not yet responded to the notification
	Not received	Notification has not (yet) been received by the DECT phone

In addition, the column headings have been largely adapted to the terms on the EM Web interface, where appropriate.

Time	Event-Id	Phase-Id	Notification-Id	Status	Source	Address	Event	Priority	Text	Location	Plan	Phase	Phase-Count	Destination	Address_	Profile	Confirmation
27.01.2025 13:48:14	2			New Event	Patient 118	118	SOS	2	Emergency Call								
27.01.2025 13:48:14	2	1		New Phase	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1				
27.01.2025 13:48:14	2	1	4	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
27.01.2025 13:48:14	2	1	5	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
27.01.2025 13:48:14	2	1	6	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	
27.01.2025 13:48:16	2	1	4	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
27.01.2025 13:48:16	2	1	5	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
27.01.2025 13:48:18	2	1	4	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	Accepted
27.01.2025 13:51:14	2	1	5	Notification terminated	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	Not confirmed
27.01.2025 13:51:14	2	1	6	Notification terminated	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	Not received
27.01.2025 13:51:14	2			Event Finished: Timeout	Patient 118	118	SOS	2	Emergency Call								
27.01.2025 14:13:45	3			New Event	Patient 118	118	SOS	2	Emergency Call								
27.01.2025 14:13:45	3	1		New Phase	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1				
27.01.2025 14:13:45	3	1	7	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
27.01.2025 14:13:45	3	1	8	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
27.01.2025 14:13:45	3	1	9	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	
27.01.2025 14:13:47	3	1	7	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
27.01.2025 14:13:47	3	1	8	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
27.01.2025 14:13:51	3	1	7	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	Rejected
27.01.2025 14:13:53	3	1	8	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	Rejected
27.01.2025 14:16:45	3	1	9	Notification terminated	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	Not received
27.01.2025 14:16:45	3			Event Finished: Timeout	Patient 118	118	SOS	2	Emergency Call								

DECT and BLE Locating

Introduction

The Event Managers Locating supplements the Event Manager functionality described above with a textual and graphical display of the position of a DECT device based on the DECT radio coverage by a base station and by Bluetooth Low Energy (BLE) as a form of wireless communication designed specifically for short-range communication. Typically the DECT radio coverage is approx. 30 to 50 meters in buildings depending on the structural conditions and approx. 300 meters in free field and for BLE beacons between 10 and 100 meters but can be differ due to environment and device classes.

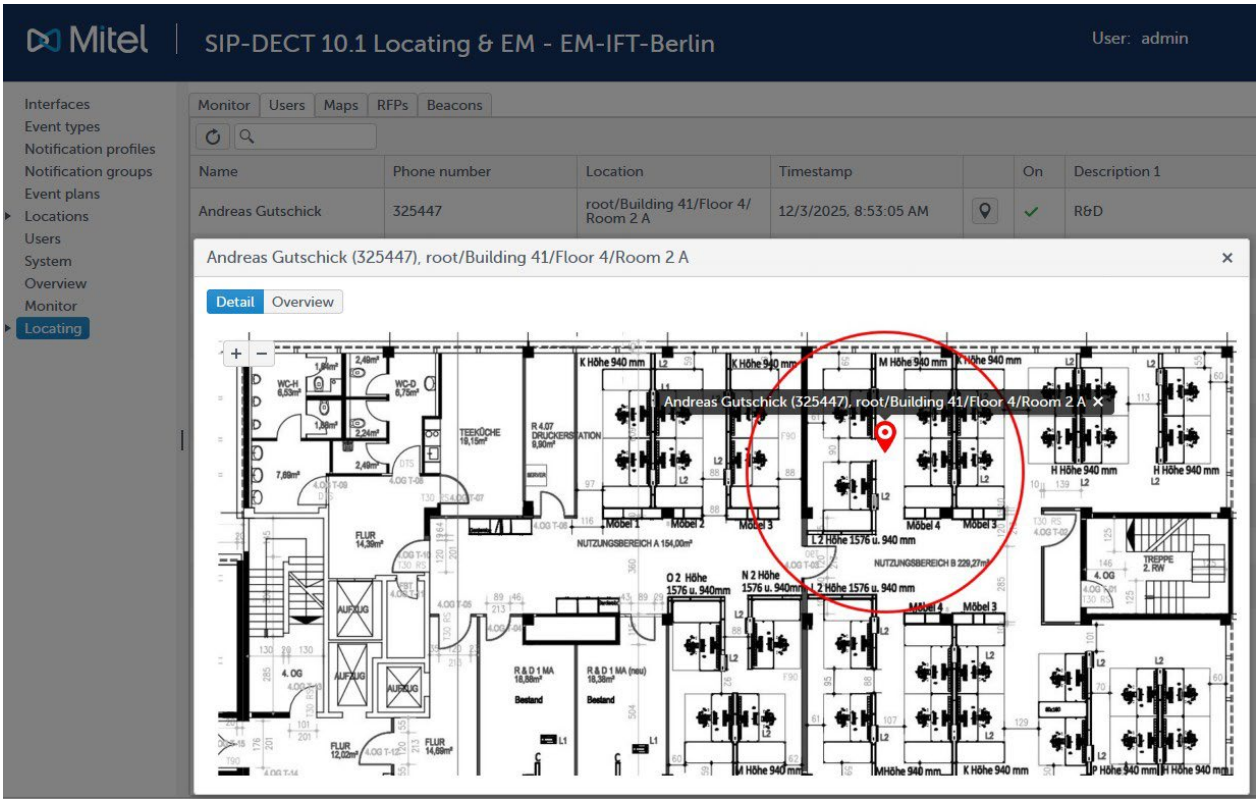
In the event of an emergency call, triggered by pressing the SOS button on the Mitel DECT telephone (722d, 732d, 742d, 632d(t) V2) or by a sensor alarm of the DECT device (732d, 742d, 632d(t) V2) as well as feature access codes for customer-specific configurable alarm triggers the location of the DECT phone will be transmitted based on the coverage of the BLE beacons (if available) or by the DECT radio coverage. In addition, the position of a locatable DECT device can also be queried independently of an event from the user list in the Locating part of the Event Manager web application.

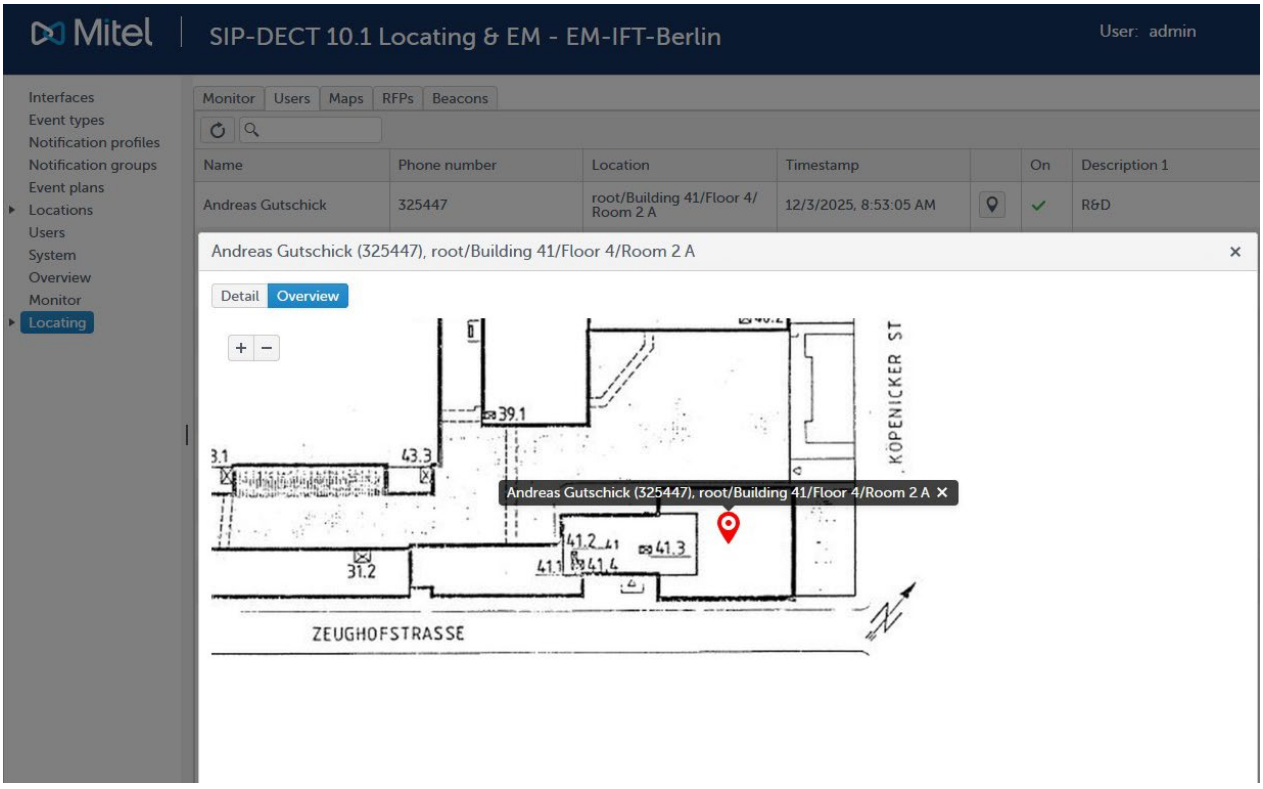
The main prerequisites for the locating application are:

- Installation of the Event Manager on a Rocky Linux server (in a Microsoft® Hyper-V server environment, a VMware® environment, or in a KVM/QEMU based virtualization environment with the installation type EM).
- Upload of a Mitel SIP-DECT Locating Server License and the Mitel SIP-DECT Locating License XXX User or Mitel SIP-DECT BLE Locating License XXX for a number of locatable DECT users in the Open Mobility Manager
- The BLE functionality on the handsets will be activated by the SIP-DECT system due to the availability of the Mitel SIP-DECT BLE Locating License XXX User.
- For DECT and BLE locating the SIP user parameter ‘Locatable’ must be set to activate the transmission of locating information from the user’s DECT telephone in the case of outgoing DECT connections and upon request by the locating application.
- The SIP user parameter “Tracking” activates the continuous and unsolicited reporting of locating information from the DECT telephone. Please note that this leads to increased DECT traffic and must be taken into account when planning the throughput. In order not to overload the DECT network, DECT phones do not send updates more frequently than every 20 seconds when locating information changes.
- The configuration requires the provision of building and floor plans in the form of graphic files (file formats “.png” and “.jpg” are supported).

Since the Mitel CloudLink daemon is not available for server installations of the Event Manager, the remote management of the Event Manager and the SIP-DECT Locating application is not available in this case.

The graphical representation is available from the locating monitor and from the locating user list in a detailed and in an overview view if the maps have been uploaded to the server and the locations have been placed on these maps.

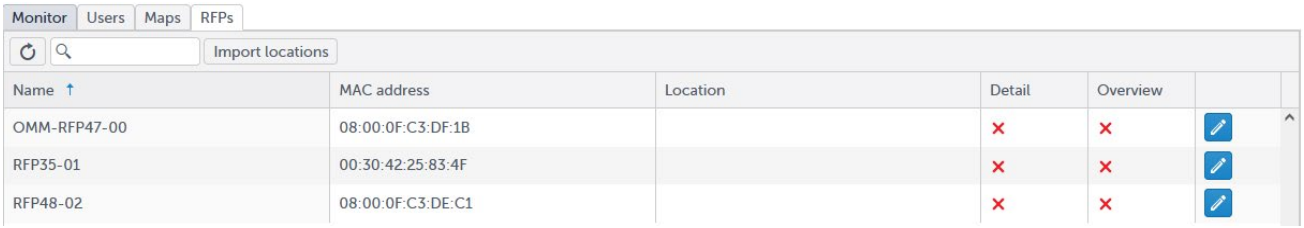




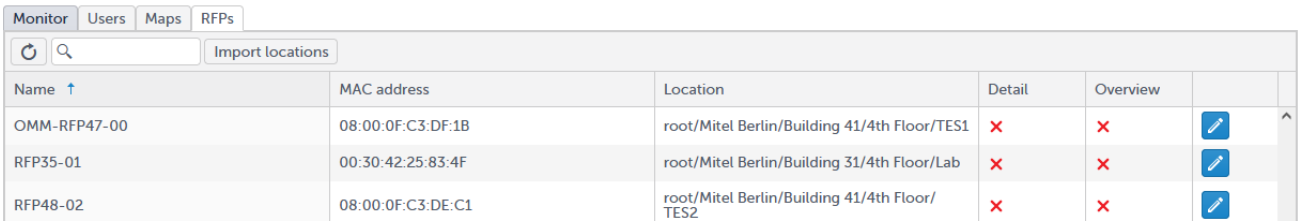
Steps for configuration of the locating application

The configuration must be executed by an administrator user of the Event Manager. The additional menu entry ‘Locating’, a new page with different tabs (Monitor, Users, Maps, RFPs, and Beacons), is available only if the Event Manager is running on a Linux server and a Mitel SIP-DECT Locating Server License is available in the connected SIP-DECT system.

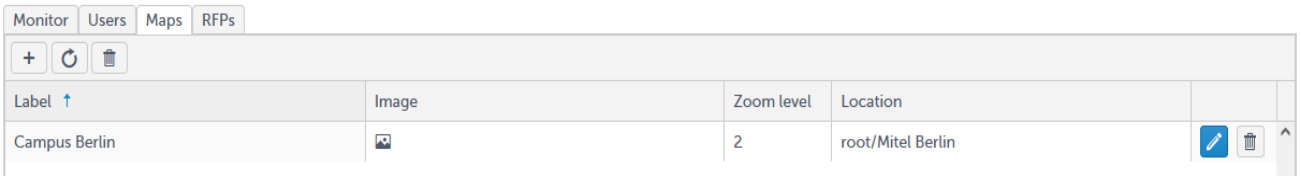
In the RFPs tab are visible all configured Radio Fixed Parts of the SIP-DECT system. They are automatically imported into the database of the Event Manager. The table includes the name and MAC address of the Radio Fixed Parts as they have been imported from the OMM.



A location can be assigned here to each of the RFPs or be imported via ‘Import locations’ button for all RFP. The result is visible in the picture below. The red crosses in the columns for ‘Detail’ and ‘Overview’ show, that these locations are positioned neither on a detail nor an overview map at this time.

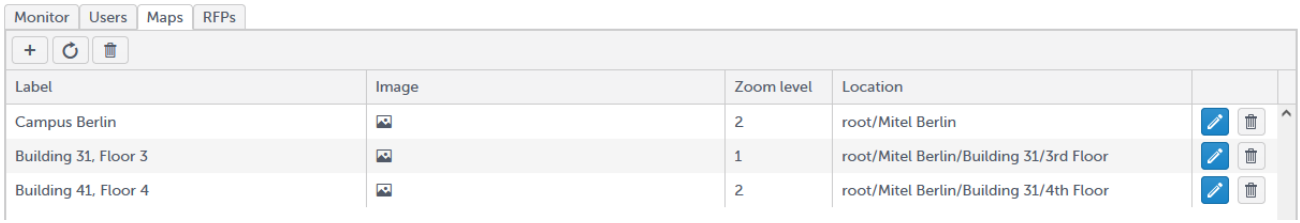


In the next step these necessary maps must be uploaded into the Event Manager via the ‘Maps’ tab. It is recommended to upload at least one overview map e.g. for the campus and as much detail maps for special floors or building parts. Supported graphic formats are PNG and JPG with resolutions of 1024, 2048, 4096 or 8192 pixel, which leads to zoom levels 1, 2, 3 or 4.



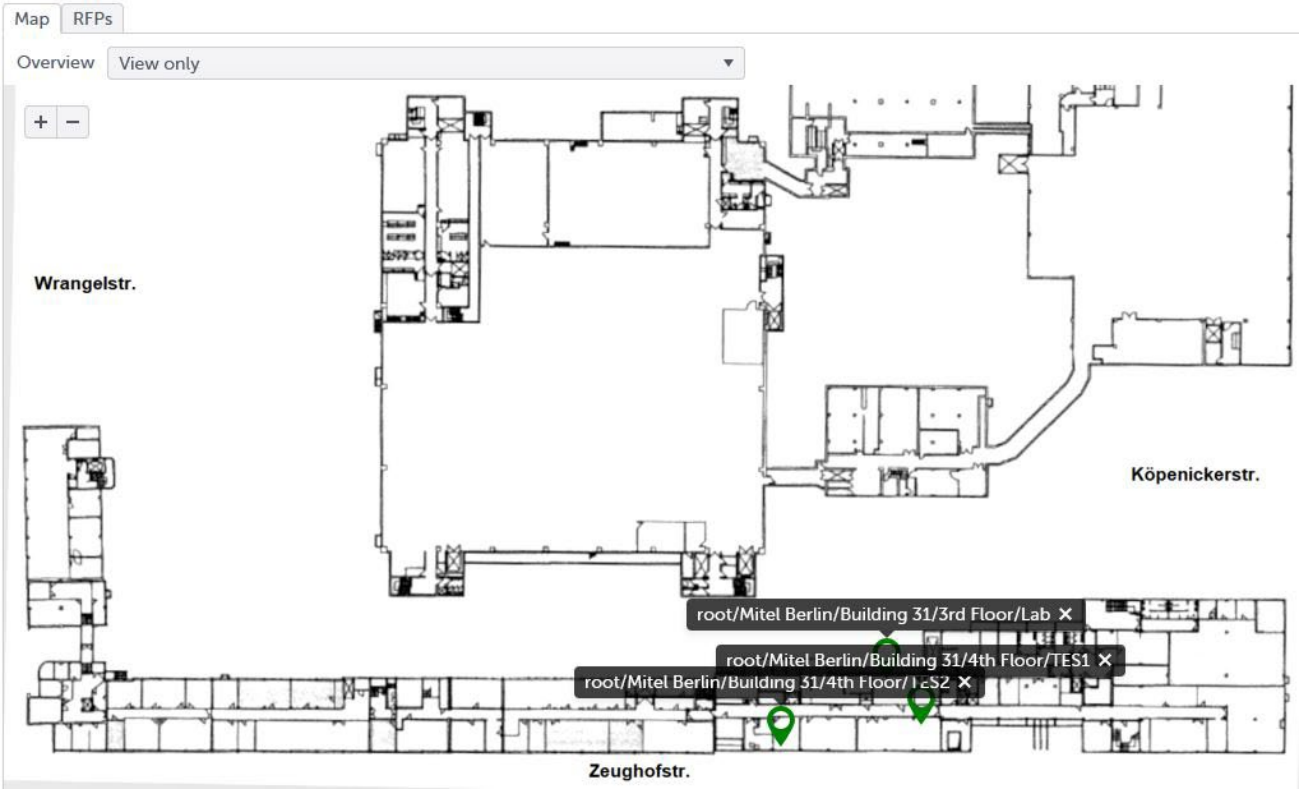
During the upload of the maps the direct assignment to a dedicated location (configured already or imported

via the RFPs tab in the step before) is possible, otherwise this must be done in a separate step after the map upload. As a result, the final table shows then all available maps with links to the uploaded images, with value of zoom levels available (depending on the resolution of the uploaded maps) and with the assigned location.

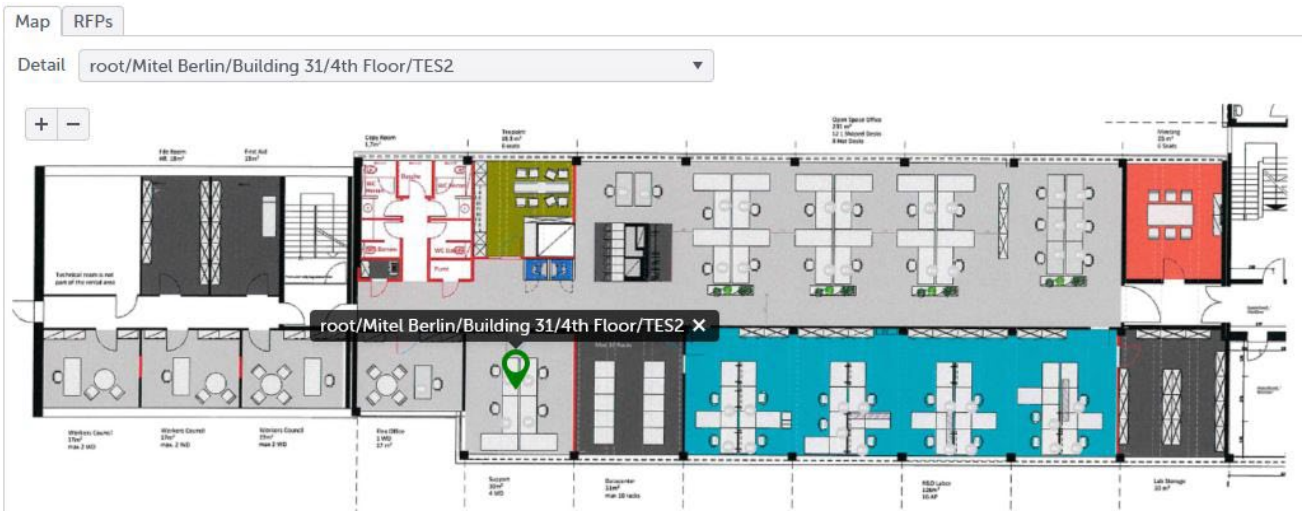


Now the locations must still be positioned on the maps. This is done in the location tree below the menu entry ‘Locating’ on the different levels (in the example from above these are the entries ‘Campus Berlin’ for the overview and ‘Building 31, 3rd Floor’ and ‘Building 31, 4th Floor’.

For the overview there must be selected the dedicated locations step by step from the dropdown menu and then the position marker must be set on the map. At the end the overview (view only) will look like the following picture.



A similar handling is needed for setting of the location positions on the detailed maps, e.g. for a special floor.



When all locations are assigned with their position marks on the detailed and overview maps the table behind the RFP tab in the ‘Locating’ menu entry will show now green ticks in the columns for ‘Detail’ and ‘Overview’ like it is visible in the following picture. If there is still visible a red cross instead of a green tick in one of the columns, this means that there is still missing the location mark on the mentioned map for the dedicated location.

The locating function of the Event Manager (EM), which was first time introduced with SIP-DECT 10.0, has been expanded in version 10.1 to include support for BLE locating.

This includes the configuration of BLE beacons and the evaluation of BLE information by the EM when determining the current position of a DECT phone. The BLE-enabled Mitel 700d DECT phones only consider BLE beacons that send the appropriate Universal Unique Identifier (UUID). By default this is the Mitel UUID “3815af41-839b-4ae7-b8e8- 8a3dfdfd23b5”. The UUID to be used can be changed using Configuration over Air (COA):

COA file example

```
UD_ConfigurationName=NonMitelUUID
BLE_UUID = “Customer UUID”
```

The Mitel 700d DECT phones with Bluetooth support (722d, 732d; and 742d) report up to 4 BLE beacons with the strongest signal strength. The Event Manager considers only BLE beacons that have a signal strength of at least -75dBm or better.

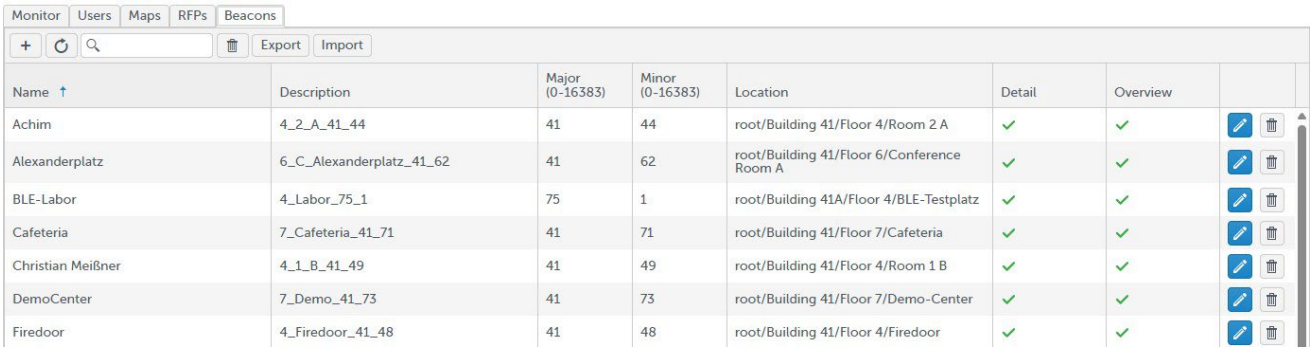
For the configuration of the BLE beacons in the EM there is available a new ‘**Beacons’** tab under the Locating pane with a list of configured BLE beacons. The BLE beacons and their locations can be imported from an Excel sheet or added via the web page separately. The Excel sheet must only include the following columns:

- Name (up to 20 characters)
- Description (up to 128 characters)

- Major (values between 0 and 16383), to be programmed into the BLE beacon
- Minor (values between 0 and 16383), to be programmed into the BLE beacon
- Location (string e.g. root/Building 41/Floor 4/Conference Room C), each part of this string (limited by '/') must be limited to 20 characters

All columns are mandatory, except the column for Location which is optional. In the case of not imported locations they must be assigned manually after the import before the BLE beacons are available for the Locating and Tracking feature.

After such an import the table looks e.g. as visible in the following picture.

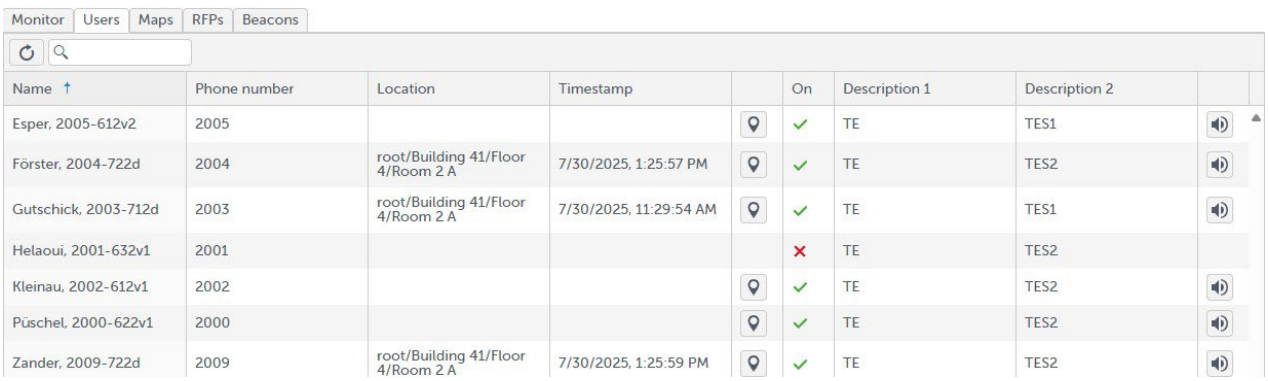


Name	Description	Major (0-16383)	Minor (0-16383)	Location	Detail	Overview
Achim	4_2_A_41_44	41	44	root/Building 41/Floor 4/Room 2 A	✓	✓
Alexanderplatz	6_C_Alexanderplatz_41_62	41	62	root/Building 41/Floor 6/Conference Room A	✓	✓
BLE-Labor	4_Labor_75_1	75	1	root/Building 41A/Floor 4/BLE-Testplatz	✓	✓
Cafeteria	7_Cafeteria_41_71	41	71	root/Building 41/Floor 7/Cafeteria	✓	✓
Christian Meißner	4_1_B_41_49	41	49	root/Building 41/Floor 4/Room 1 B	✓	✓
DemoCenter	7_Demo_41_73	41	73	root/Building 41/Floor 7/Demo-Center	✓	✓
Firedoor	4_Firedoor_41_48	41	48	root/Building 41/Floor 4/Firedoor	✓	✓

When all locations have the green ticks in the columns 'Detail' and 'Overview', the configuration is completed. In this case the 'Users' tab in the 'Locating' menu entry will now show the completed list of all those SIP-DECT users that are configured in the SIP-DECT system at least with the 'Locatable' attribute and probably also with the 'Trackable' attribute.

The table will show the attributes 'Name, 'Phone number', 'Location' (only for trackable users), a graphical link to the location map, the status of the user's phone, and also the timestamp of its last action together with two description fields imported from the SIP-DECT system including information like department or team. Starting with SIP-DECT 10.1 there is available an additional icon for requesting the so called Locating Alert for those users which are locatable and from which the Event Manager have received already their latest known location.

Therefor the portable parts at least must be switched on, otherwise they would be marked with a red cross in the column "On". Users without the "Trackable" feature activated in SIP-DECT will be shown without an entry in the location column.



Name	Phone number	Location	Timestamp		On	Description 1	Description 2	
Esper, 2005-612v2	2005				✓	TE	TES1	
Förster, 2004-722d	2004	root/Building 41/Floor 4/Room 2 A	7/30/2025, 1:25:57 PM		✓	TE	TES2	
Gutschick, 2003-712d	2003	root/Building 41/Floor 4/Room 2 A	7/30/2025, 11:29:54 AM		✓	TE	TES1	
Helaoui, 2001-632v1	2001				✗	TE	TES2	
Kleinau, 2002-612v1	2002				✓	TE	TES2	
Püschel, 2000-622v1	2000				✓	TE	TES2	
Zander, 2009-722d	2009	root/Building 41/Floor 4/Room 2 A	7/30/2025, 1:25:59 PM		✓	TE	TES2	

The content of the table will be updated automatically due to actions of the user's phones and, in case of

“trackable” users, when the locating information of the DECT phone changes.

In the ‘Monitor’ tab of the ‘Locating’ menu entry additionally to the ‘normal’ monitor will be visible a location link in case of an event generated by a locatable user, e.g. in case of SOS key or Man Down alarm initiated by a SIP-DECT phone, and a Locating Alert link.

Locating Alert

Starting with release 10.1 of the SIP-DECT Event Manager the system offers the ability of starting an ‘Locating Alert’ to a dedicated DECT portable part, which is used by a locatable SIP-DECT user. The ‘Locating Alert’ request can be started either from within the Monitor tab or from the Users tab in the Locating pane by those users for whom an endpoint in the new Web event interface is configured. The notification profile which is used for this notification to the portable part is not configurable but fixed coded with increasing alerting volume, priority 5, white foreground color for the notification text and red background color. This type of notification is only triggerable once at a time in the whole system and could not be overwritten by other notifications to the same endpoint.

Backup and restoring the Event Manager data including the installed graphic files

The Event Manager database, which can be backed up and restored via the EM web service, does not contain the uploaded graphic files for locating, as the graphic files may be very large.

However, as there is a dependency between the configuration data and the graphic files, these must be backed up and restored together, e.g. when transferring an existing configuration to a new installation.

In addition, the EM application should not be running during the backup and restore process to avoid creating unwanted inconsistencies through parallel activities.

So that these processes do not have to be carried out manually, the Event Manager automatically provides two shell scripts that allow the simple creation of a complete backup and restore. Both scripts must be executed on the command line interface by the root user.

The script `sip-dect-em-create-backup.sh` is used to create a data backup. The script requires as an argument a target directory in which the data backup is to be stored. This directory must already exist. The generated file will then have the name sip-dect-em-backup_<timestamp>.tar.gz with the current timestamp from date and time e.g. 20250121_162259. During the execution of the script the sip-dect-em service will be

```
[root@deberrndws5090 10.0]$ sip-dect-em-create-backup.sh  /root/Downloads/
User: root
OK, you are root
check service sip-dect-em:
active
The service 'sip-dect-em' is running. Would you like to stop it? (y/Y): y
Service sip-dect-em successfully stopped.
Create achive: /root/Downloads//sip-dect-em-backup_20250121_162259.tar.gz
…
Achive /root/Downloads//sip-dect-em-backup_20250121_162259.tar.gz created
Start service sip-dect-em
[root@deberrndws5090 10.0]$
```

terminated, the user is asked for confirmation again to prevent accidental termination. After completion of creating the backup the the sip-dect-em service is restarted automatically.

The script `sip-dect-em-restore-backup.sh` is used to restore a data backup. The script requires the name of the backup file as the first argument and a target path as the second. The target path is always the root directory / unless you want the backup to be unpacked at a different location.
To back up the data in the long term, it is recommended to copy the data backup to an external backup medium. As no other services are pre-installed on the secure virtualization images, an external copy protocol client, for example SCP, must be used for this.

```
[root@deberrndws5090 10.0]$ sip-dect-em-restore-backup.sh /root/Downloads/sip-dect-
em-backup_20250121_162259.tar.gz /
User: root
OK, you are root
OK. Unpack file /root/Downloads/sip-dect-em-backup_20250121_162259.tar.gz to target
directory /
check service sip-dect-em:
active
The service 'sip-dect-em' is running. Would you like to stop it? (y/Y): y
Service sip-dect-em successfully stopped.
retore backup from /root/Downloads/sip-dect-em-backup_20250121_162259.tar.gz to /
OK. Unpack file /root/Downloads/sip-dect-em-backup_20250121_162259.tar.gz to target
directory /
…
Start service sip-dect-em
[root@deberrndws5090 10.0]$
```

Quick Start Configuration Guide SIP-DECT Event Manager

The following steps need to be followed to get a basic working configuration. There are two basic scenarios.

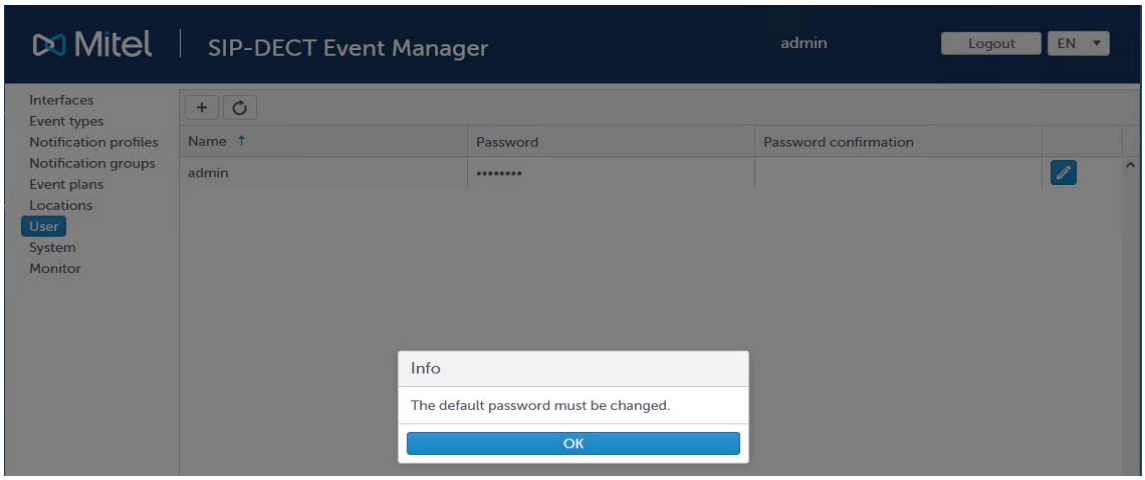
- Configuring a SOS alarm trigger from a SIP-DECT phone
- Configuring an ESPA message

The prerequisite for the following steps is a functioning SIP-DECT installation with several Mitel SIP-DECT 602d v2 / 700d terminals. The SIP-DECT terminals are already updated to the SW provided with the SIP-DECT system.

The SIP-DECT Event Manager was started on an RFP using the OM Configurator (OMC) and has the default configuration.

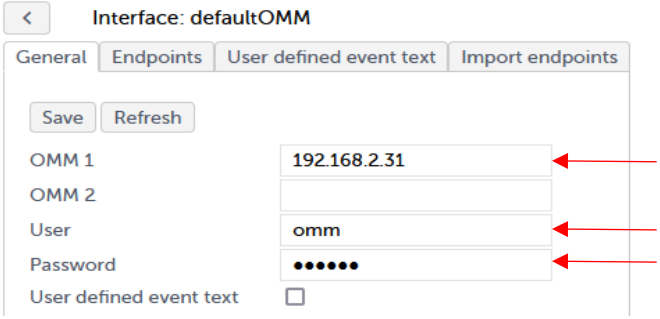
Configuring SOS alarm trigger from a DECT phone

1. Log in to the SIP-DECT Event Manager web service https://<RFP IP address>:8444 with default login “admin” and password “admin”.
2. Change the default password.



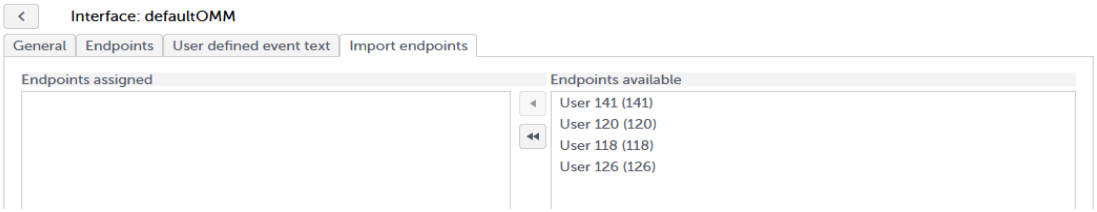
3. Open OMM interface configuration dialog by selecting the link as shown below.

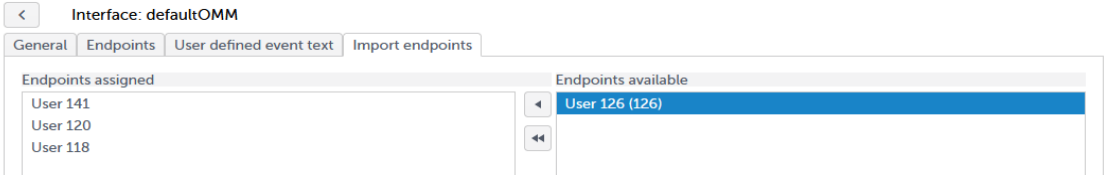
4. Enter the OMM IP address(es), user and password and confirm with ‘Save’. Return to the interface overview by clicking the Back button .



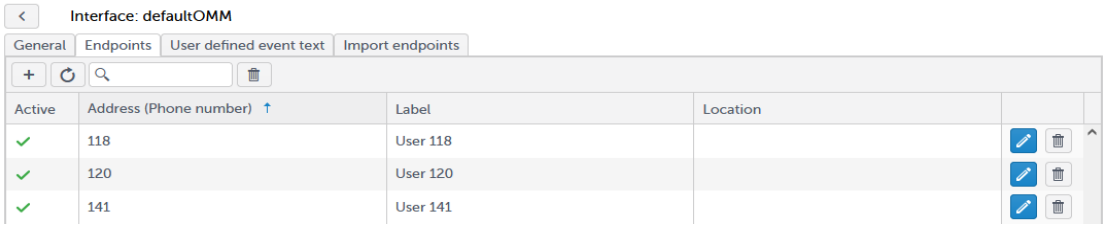
5. The interface status should change to green, indicating that the SIP-DECT Event Manager could connect with the OMM.

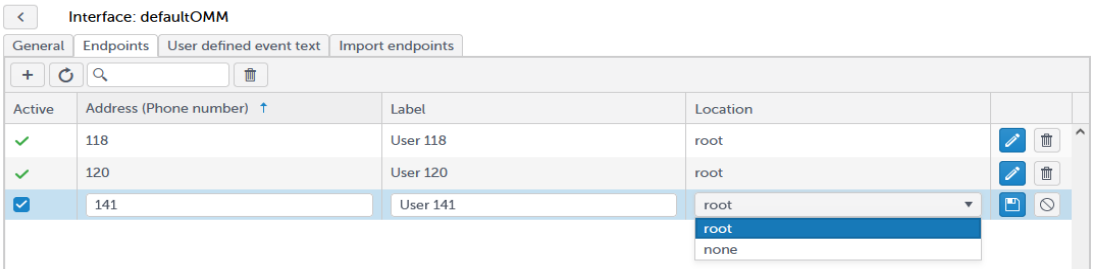
6. Go back into the OMM interface configuration dialog, click the Import endpoints tab and transfer the SIP-DECT endpoints into the SIP-DECT Event Manager configuration by selecting one by one and clicking ◂ or all by clicking ◂◂. As a result the endpoints should now appear in the endpoints list



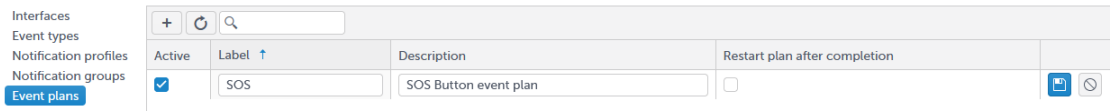


7. Assign the endpoints to the default location root as shown below.

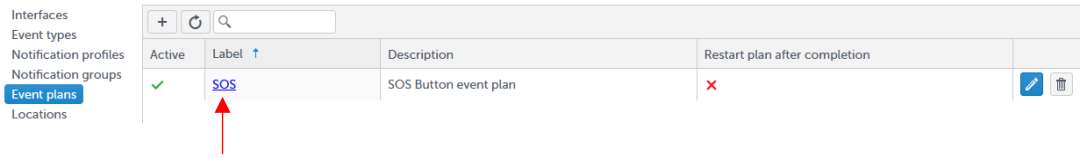




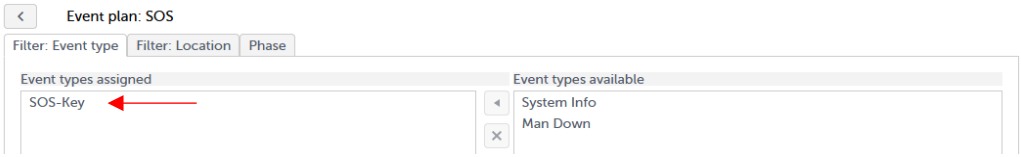
8. Click the **Event plans** configuration pane and create a new event plan by clicking +. Set a name and description for the plan and confirm with 💾.



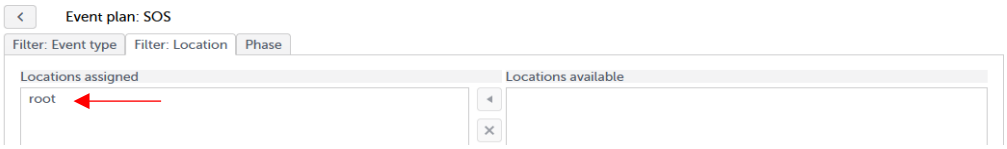
9. Click on the newly created plan.



10. Under the **Filter: Event type** tab, add the default event type SOS-Key to the event type filter.



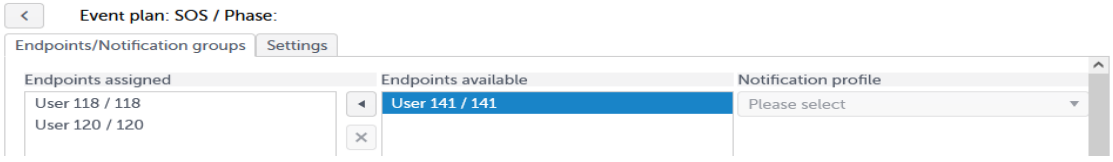
11. Click **Filter: Location** tab and add the default location root to the location filter.



12. Click the **Phase** tab and create a phase for the event plan by clicking New. Set a name and description for the phase and confirm with .

13. Open the Phase configuration dialog by selecting the link as shown below.

14. Transfer the endpoints you want to be notified into the endpoints list by selecting one by one (to select more than one use additionally the Ctrl key) and press . The default notification profile ‘Normal’ will automatically be assigned.



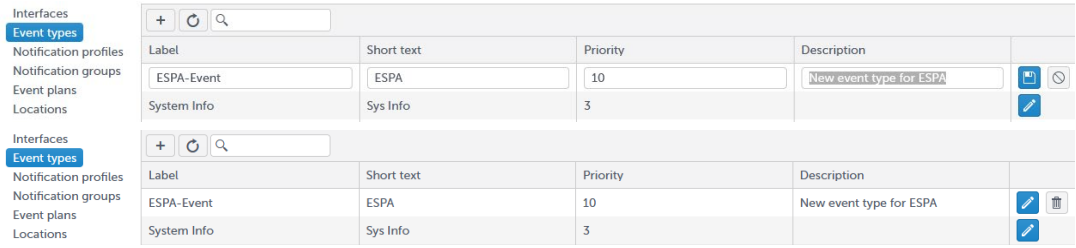
No further phase settings need to be changed. Return to the main level dialog by .

15. If now the SOS button is pressed on one of the Mitel SIP-DECT phones, a notification should appear on those SIP-DECT terminals that have been assigned as endpoints to the event plans phase.

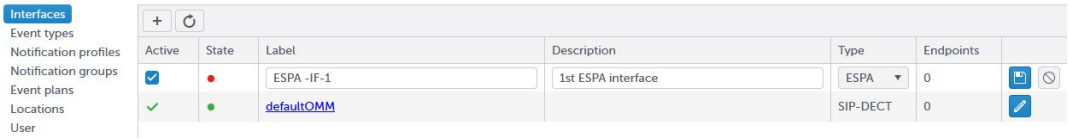
Configuring an ESPA interface

Execute the same steps to setup the ESPA interface, add the ESPA interface endpoints and assign the default location “root” as described in the Configuring SOS alarm trigger from a DECT phone section. Before a new event plan is created, the ESPA interface must be set up and a new event type must be created.

1. Click the **Event types** configuration pane.
2. Add a new entry by clicking +. Set a unique label and short text and confirm with .

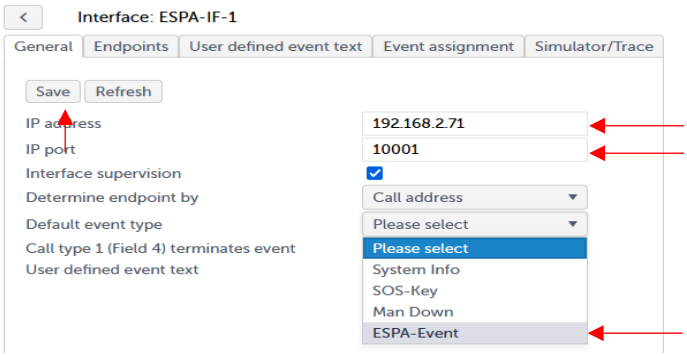


3. Click the **Interfaces** configuration pane.
4. Add a new entry by clicking +. Set a unique label and description and confirm with . Ensure that the interface type ‘ESPA’ is selected under ‘Type’

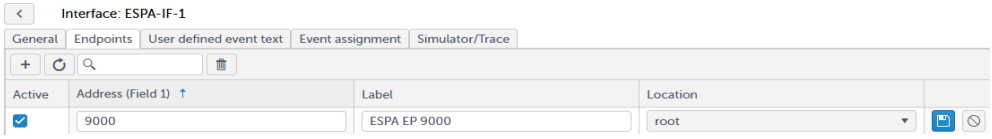


5. Open the Interface configuration dialog by selecting the created link.

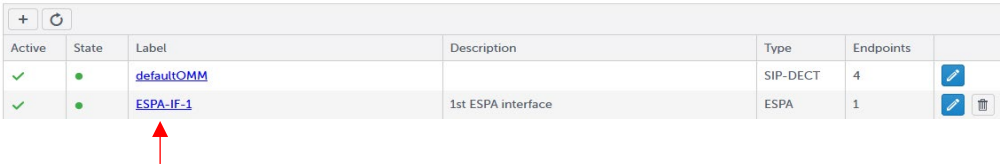
6. Enter the IP address and port that the ESPA 4.4.4 of the SIP-DECT Event Manager should connect to, select the Default event type and confirm with Save.



7. Add an ESPA endpoint in the **Endpoints** tab. Set the endpoint address (ESPA field 1 – Call address), assign a name and the default location 'root' and confirm with

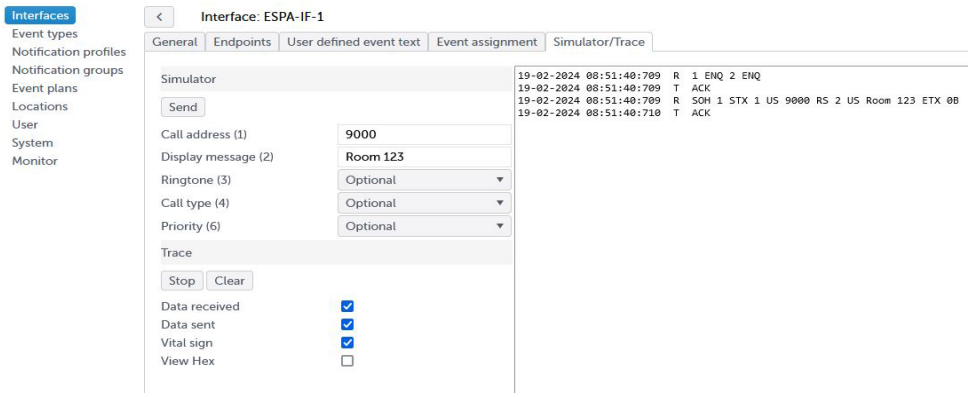


8. Return to the interfaces overview by clicking . If the SIP-DECT Event Manager could connect with the nurse call system or similar the interface status turns to green

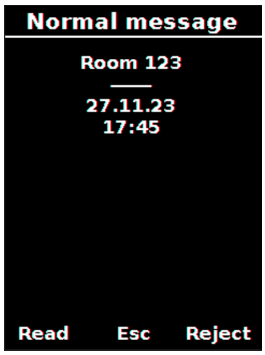


9. Create an event plan. Follow steps 8-15 as described in the Configuring SOS alarm trigger from a DECT phone section. However, this time the newly created event type should be assigned to the ESPA interface as the default event type to use.

10. To trigger an event even without a connected system, there is useable the simulator function of the ESPA interface



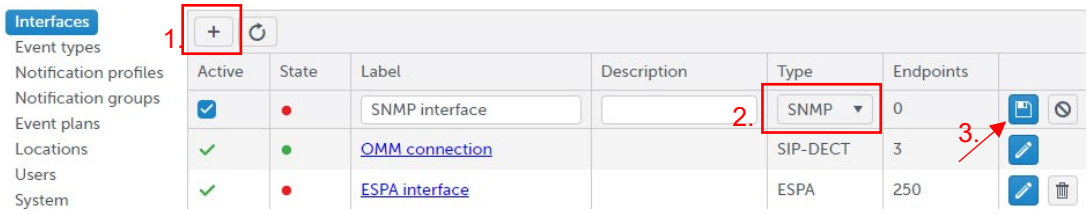
11. When an ESPA message is received, a notification with the received text message should now appear on the Mitel SIP-DECT terminal assigned to the event plans phase.



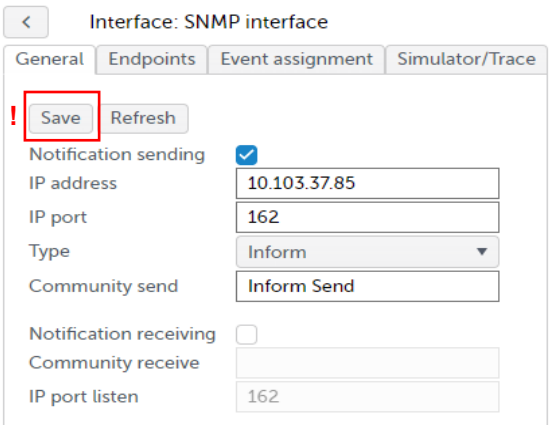
Configuring an SNMP interface

This chapter will explain how to configure an SNMP interface to send and receive Traps & Inform-Requests step-by-step. Before following this guide, make sure to have a working SIP-DECT interface with endpoints. As an example for a Trap sender, whose notifications the Event Manager receives and processes, we will be using the Inveo Nano Temperature Sensor.

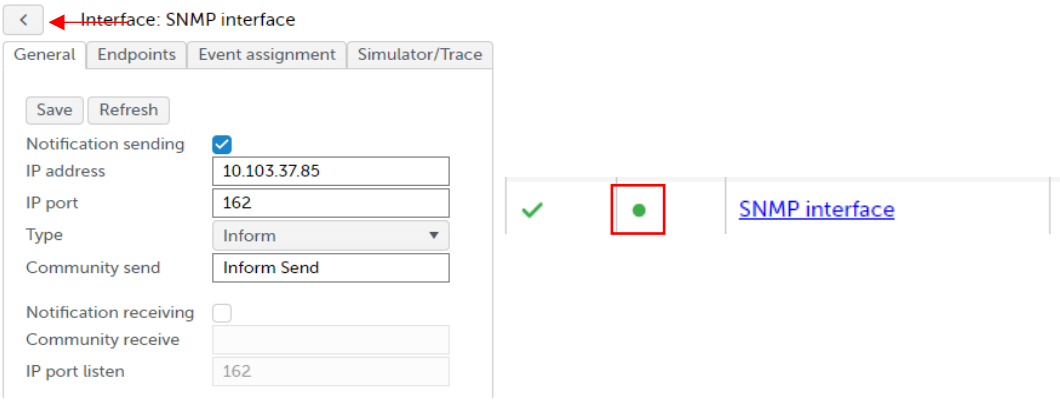
1. Enter the “Interfaces” dialogue. Then create and name a new SNMP interface. Make sure the interface is set to active.



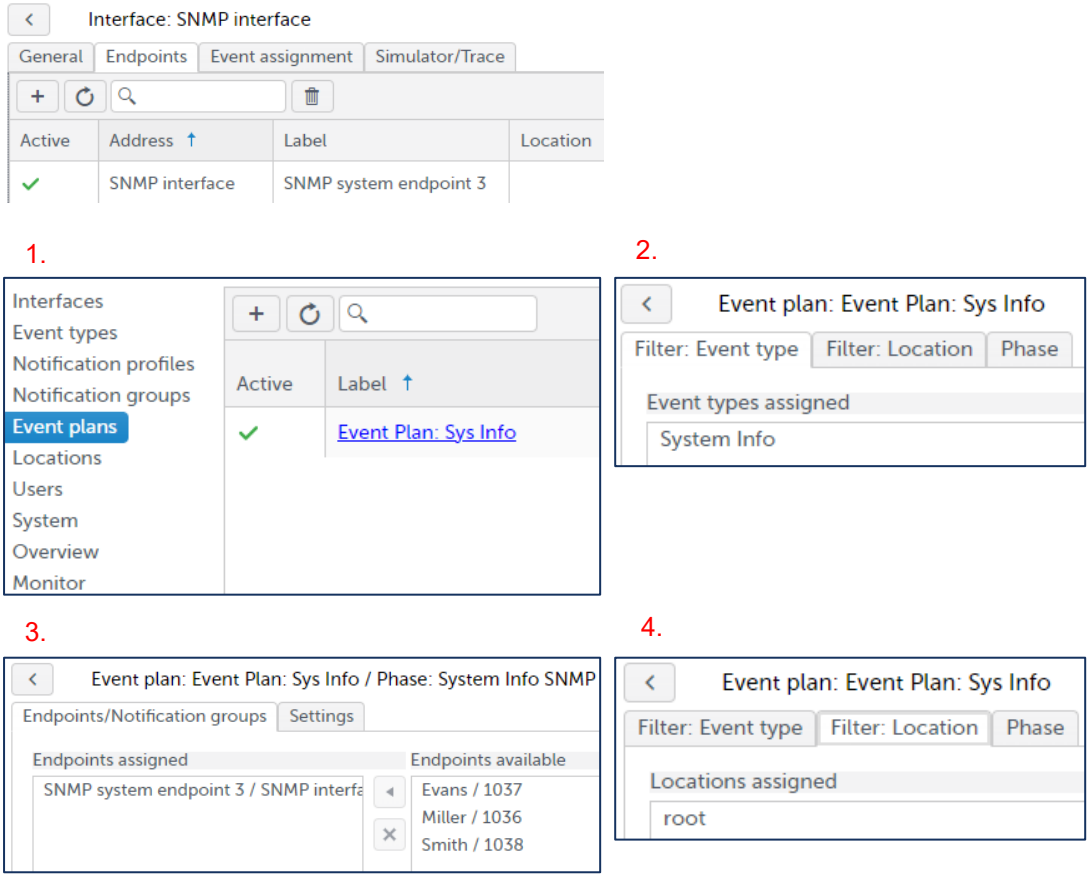
2. Click onto the newly created SNMP interface’s name. You should now be inside the tab “General”. Tick the “Notification sending” box and enter the IP address and IP port of the trap receiver you wish to send traps to into their corresponding fields. Select if you want to send Inform-Requests or simple Traps in the dropdown menu “Type” and proceed to enter a valid community string in the field “Community send”. Make sure the box “Notification receiving” is unchecked for now. Press the button “Save” at the top left.



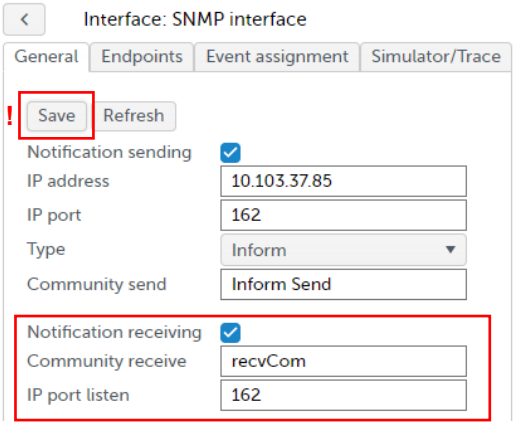
3. Click the arrow at the top left. You should now be back in the “Interfaces” dialogue from the very start. Make sure the SNMP interface’s status is now active (green). If you did something wrong or forgot to save, it should either be red (inactive) or yellow (misconfigured). If it is green, proceed with the next instructions. If it is any other color, repeat 2.



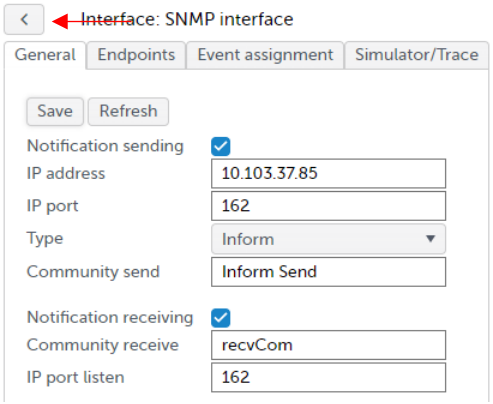
4. After clicking onto the SNMP interface’s name and entering its configuration window, click onto the tab “Endpoints”. An Endpoint with the label “SNMP system endpoint X” (with X being a number) should be there and it should be active. If it is not active, check if you have ticked “Notification sending” inside the “General” tab. If you want the SNMP interface to send Traps/Inform-Requests, you will need to add this system endpoint into an event plan’s phase. When that event plan is triggered by an event and it reaches the phase with the SNMP system endpoint in it, the interface will send a notification to its configured destination. In this example, we are adding the SNMP system endpoint into an event plan in the location “root” which gets triggered by the predefined event type “System Info”. This results in our SNMP interface sending interfaceStatusChange notifications when any interface changes its status.

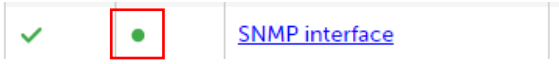


5. Next, we will configure notification receiving and processing. To start off, we will once again enter the “General” tab of our SNMP interface. Tick the box “Notification receiving”, enter the community string we expect to receive in the text field “Community receive” and enter the IP port this SNMP interface is supposed to listen for notifications on into the field “IP port listen”. Press “Save”.

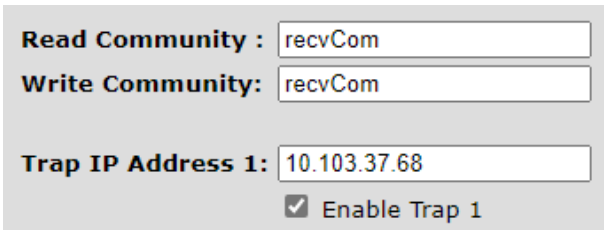


6. Leave the “General” tab through the arrow at the top left. In the interface overview window, check if the SNMP interface is still active (green). If it is active, proceed with the next set of instructions. If it is red (inactive) that means the IP port you tried to configure for listening is already taken by some other process or interface. It cannot be used. Re-enter the SNMP interface’s configuration, select a different port and press “Save”. Recheck the interface’s status. Repeat until the SNMP interface is active (green).



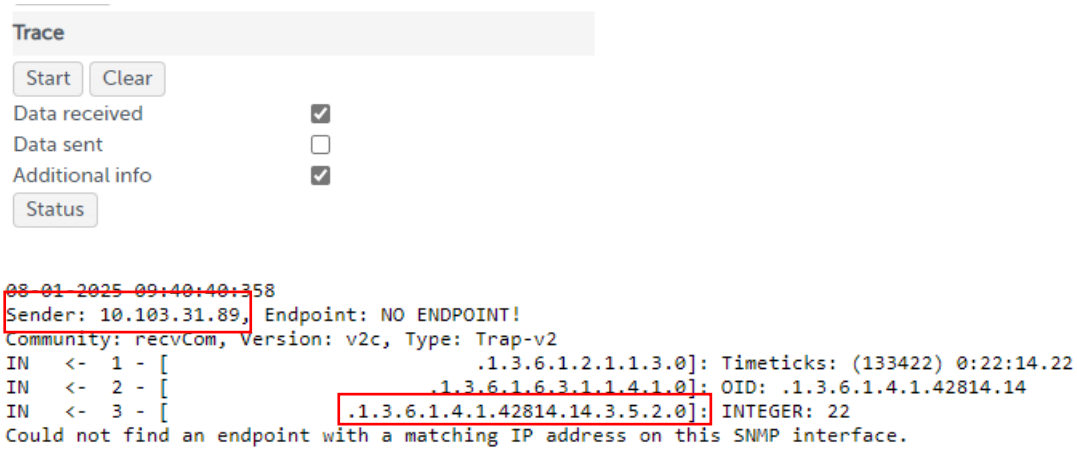


7. Now, configure the device you wish to receive notifications from to be able to send Traps/Inform-Requests to the Event Manager's SNMP interface. In this example, we are configuring the Inveo Nano Temperature Sensor to send Traps. This step may look vastly different in your use case with your device. Please follow the instructions of the manufacturer of the device you are configuring and ask them for help if you encounter any problems. Make sure that the trap community that the sending device sends, is the same as that one the Event Manager's SNMP interface has configured in the field "Community receive".

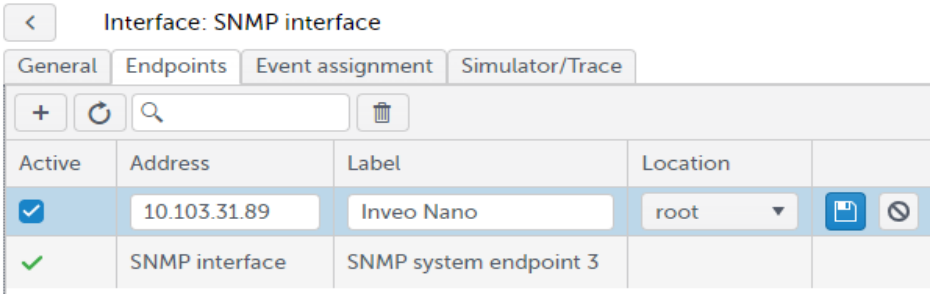


8. In order to process the received SNMP notifications, an Endpoint with the sender's IP address needs to be created as well as an Event assignment which reacts to the correct Object Identifier (OID). If you already know the IP address of your SNMP notification sender and what OIDs it is sending in its notifications, you may skip step 9.

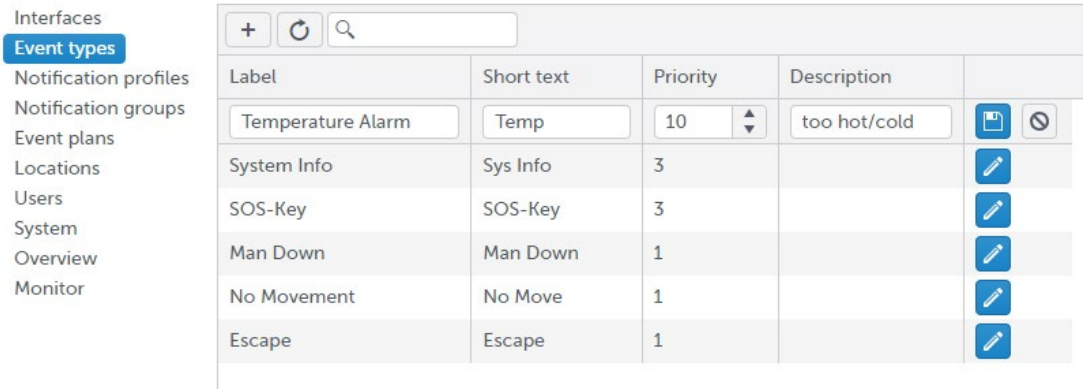
9. To process received SNMP notifications an SNMP endpoint with the sender's IP address as well as a matching event assignment need to be created. In order to figure these out easily, enter the SNMP interface's tab "Simulator/Trace". Tick the boxes for "Data received" and "Additional info" and untick the box "Data sent" at the very bottom under the headline "Trace". Now press "Start". The trace window to the right will now display any and all incoming notifications on this interface. In order to figure out the sending devices' IP address as well as the OIDs it is supplying in its sent SNMP notifications, make it send a notification to the Event Manager, read out the displayed IP address and decide which OID you wish to assign an event to. The event manager is incapable of knowing what any received Object Identifier means. This information is contained inside the MIB files of the notification sending device and need to be read out on your own. In this example, the OID ".1.3.6.1.4.1.42814.3.5.2.0" contains the current temperature, which is sent by the Inveo Nano Temperature Sensor if it is too hot or too cold according to its configuration. Once you are done, press "Stop" to deactivate the trace functionality.



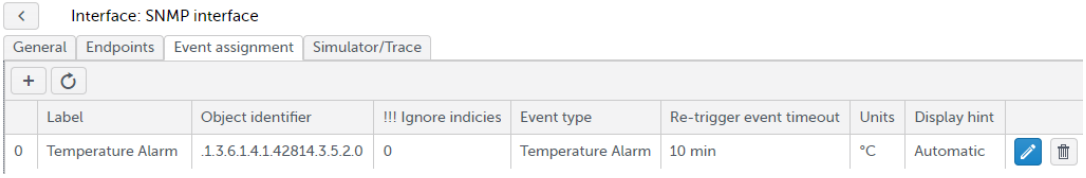
10. Now that we have the sender's IP address, we will create an SNMP endpoint with the IP in the "Address" field and an easily recognizable label. This can be done inside the "Endpoints" tab inside the SNMP interface. Furthermore, we will assign it the location "root". You may add it to a different, more fitting location.



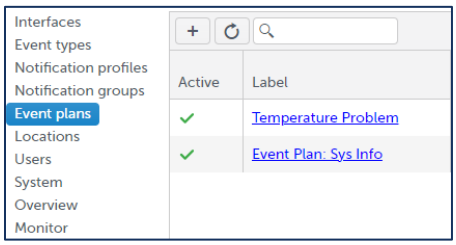
11. Create a new event type which fits whatever information you are receiving from the SNMP notification sender.

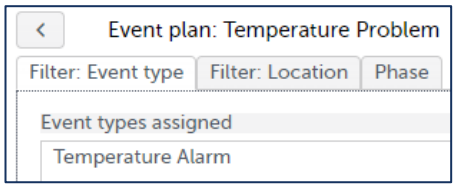


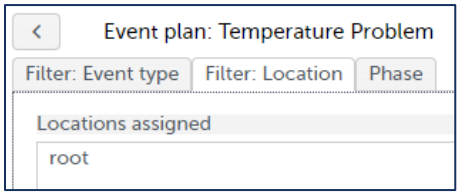
12. Create an event assignment with the correct Object Identifier. Note: The first 2 OIDs of an SNMPv2 notification are the same in all SNMPv2 notifications. Creating an event assignment that matches with the first 2 OIDs (**.1.3.6.1.2.1.1.3.0** & **.1.3.6.1.6.3.1.1.4.1.0**) will match with ALL correct v2 notifications. Since the first matching event assignment is chosen, this will lead to all SNMP notifications triggering the same event.

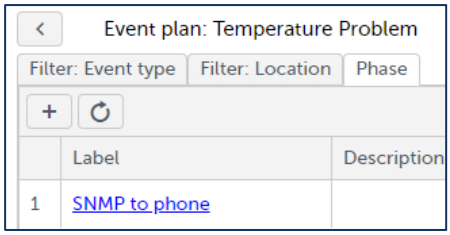


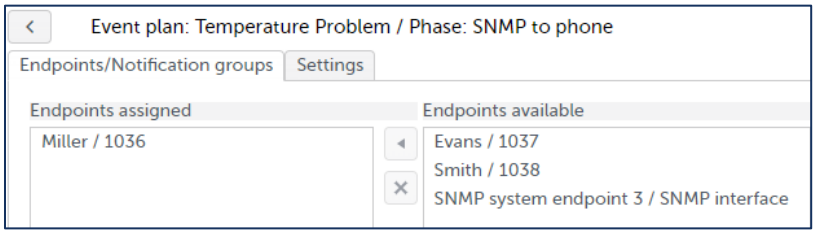
13. Add an event plan into the location you assigned the endpoint to (“root” in this example). This event plan should react to the event type you used inside the “Event assignment” tab of the SNMP interface. Add a phase to the event plan as well as SIP-DECT phones as recipients inside said phase.

1.


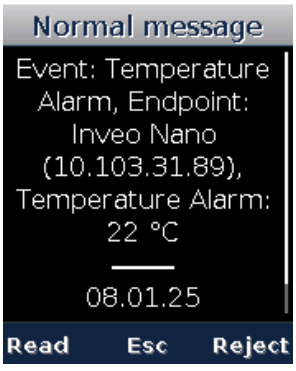
2.


3.


4.


5.


14. Once the SNMP notification sender sends a trap/inform-request our way and everything was set up correctly, the following message should be displayed inside SIP-DECT phones assigned to a phase of the newly created event plan.



15. In case this did not happen, you may enter the SNMP interface’s “Simulator/Trace” tab, tick the boxes “Data received” and “Additional Info” under “Trace” and start the trace. Once an SNMP notification has been received, a message will be displayed after the received notification, telling you what ended up happening while processing the

notification. This may indicate what went wrong while setting up the SNMP interface. If the message says that everything went well, but you still do not see a message inside the desired SIP-DECT phone, the problem may lie inside the SIP-DECT interface or within the event plan you set up.

Configuring an IP Phone interface

This chapter will explain how to configure an IP Phone interface to receive and trigger event step-by-step. Before following this guide, make sure to have a working SIP-DECT interface with endpoints. As an example phone, we will be using the both the Mitel MINET 6930 as well as the Mitel 6930 SIP phone.

Configuring a MINET phone

A prerequisite for sending and receiving events to and from the Event Manager is a working and configured MINET phone running on a MiVoice Business.

1. Enter the “Interfaces” dialogue. Then create and name a new IP Phone interface. Make sure the interface is set to active.

2. Decide whether or not the IP Phone interface should validate the certificates on incoming connections. When a phone sends any request to the Event Manager, the phone’s certificates will be checked and the request will be discarded should the certificates not be trusted. Only incoming connections are checked. The Event Manager will not validate certificates of the recipient when sending notifications to a phone.

3. Enter the “Endpoints” tab inside the interface and configure your endpoint. The address field contains the case sensitive SIP username, the label is freely configurable, the column IP address cannot be edited and will be filled in automatically once the MINET phone polls the Event Manager and the location is to be configured however needed.

4. Enter the “URIs” tab. Locate “Config file for MiVoice Business” and select the appropriate 6900 phone type. In our case that would be 6930. If certificate validation is off, use the one under “Without certificate validation. Otherwise, use the one under “With certificate validation”. Press download in order to get a .cfg file for that phone type.

5. Now enter your MiVB System Administration Tool. Go to “Users and Devices” → “Advanced Configuration” → “Phone Application Update”, press “Upload Application”, upload the recently downloaded .cfg file and select all phones/users that should connect to the Event Manager over the IP-Phone interface. Press upload. All selected phones will now occasionally poll the Event Manager and are now able to receive notifications.

6. In order to configure a key to trigger events from a MINET phone, enter the “URIs” tab and copy the needed “Event” URI. This choice once again depends on if certificate validation has been toggled on or off.

7. Enter the MiVoice Business System Administration Tool and go to “Users and Devices” → “User and Services Configuration”. Select the service of the MINET phone you wish to add an event triggering key to. Enter the tab “Keys”. Select a key, give it a fitting label, select the line type “URL Line” and paste the “Event” URI into the URL field.

8. The XML URI is still incomplete. You now must add a case sensitive event type’s name at the end of the URI that exists in the Event Manager. This event type will be triggered in the location of this SIP phone. You may also add custom event text or callback numbers to this URI. Between 2 URI parameters there must always be a ‘&’ and after the parameter’s name there must be a ‘=’ to give it its associated value. The final URI may look something like this (parameter names are underlined, parameter values are freely configurable):

   ```
   https://x.x.x.x:8444/ipphone/v1/paging?request=event&sipusername=
   ```

```
$$SIPUSERNAME$$& eventtrigger=SOS-Key&eventtext=my own SOS event
text&callback=1234
```

9. Configure a 2nd button in the same manner with the “Event 2” URI in order to support redundancy. Press “Save Changes”.

10. Once you press the configured softkey on the phone’s screen, the event will be sent to the Event Manager. Should the event exist, an associated event plan be configured and the phone be a valid endpoint inside the IP-Phone interface, an event will be successfully executed. The phone will receive visual feedback if that happens in form of either a green checkmark or a red X.

Configuring a SIP phone

A prerequisite for sending and receiving events to and from the Event Manager is a working and configured SIP phone.

1. Enter the “Interfaces” dialogue. Then create and name a new IP Phone interface. Make sure the interface is set to active.

2. Decide whether or not the IP Phone interface should validate the certificates on incoming connections. When a phone sends any request to the Event Manager, the phone’s certificates will be checked and the request will be discarded should the certificates not be trusted. Only incoming connections are checked. The Event Manager will not validate certificates of the recipient when sending notifications to a phone.

3. Enter the “Endpoints” tab inside the interface and configure your endpoint. The address field contains the case sensitive SIP username, the label is freely configurable, the column IP address cannot be edited and will be filled in automatically once the SIP phone polls the Event Manager and the location is to be configured however needed.

4. Enter the “URIs” tab. If you enabled certificate validation in your IP phone interface’s general settings, press “Copy” on the row “Poll” in the section “With certificate validation”. If you did not enable certificate validation, press “Copy” on the row “Poll” in the section “Without certificate validation”.

5. Now enter the SIP phone’s web configurator. Login and select the tab “Action URI”. Paste the copied “Poll” URI in any of the empty Poll URI fields. It does not matter which number is chosen. Set the interval to 30. Press “Save Settings”.

6. Repeat steps 4 & 5 for the “Poll 2” URI if it exist. This is the IP address to the redundant Event Manager. Without configuring this URI, the SIP phone will become unreachable by the Event Manager in case of a failover.

7. Enter the tab “Configuration Server” in the SIP phone’s web configurator. Locate the field “XML Push Server List (Approved IP Addresses)”. Enter the Event Manager’s as well as the redundant Event Manager’s IP address into that field. Separate the IP addresses with a comma. You may add more IP addresses for other system that push XML content to the SIP phone. Note however that independent systems that push XML content to phones may result in notifications from the Event Manager to get deleted from the phone’s screen. Press “Save Settings”.

8. Enter the Event Manager’s IP Phone interface configuration once more. Enter the tab “Endpoints”. Once the SIP phone polls the Event Manager, its IP address should be shown in the column “IP address”. This SIP phone is now capable of receiving and displaying events on its screen.

9. In order to configure a key to trigger events from a SIP phone, enter the “URIs” tab and copy the needed “Event” URI. This choice once again depends on if certificate validation has been toggled on or off.

10. Enter the SIP phone's web configurator once again. You may access it normally or by clicking on the IP address shown in the endpoints tab.

11. Enter the tab "Softkeys and XML". Select the key you wish to configure on the phone. Select "XML" as its type, give it a fitting "Label" and paste the "Even" URI into its "Value" field.

12. The XML URI is still incomplete. You now must add a case sensitive event type's name at the end of the URI that exists in the Event Manager. This event type will be triggered in the location of this SIP phone. You may also add custom event text or callback numbers to this URI. Between 2 URI parameters there must always be a '&' and after the parameter's name there must be a '=' to give it its associated value. The final URI may look something like this (parameter names are underlined, parameter values are freely configurable):

    `https://x.x.x.x:8444/ipphone/v1/paging?request=event&sipusername=$$SIPUSERNAME$$& eventtrigger=SOS-Key&eventtext=my own SOS event text&callback=1234`

13. Configure a 2nd key in the same manner with the "Event 2" URI in order to support redundancy. Press "Save Settings".

14. Once you press the configured key on the phone, the event will be sent to the Event Manager. Should the event exist, an associated event plan be configured and the phone be a valid endpoint inside the IP Phone interface, an event will be successfully executed. The phone will receive visual feedback if that happens in form of either a green checkmark or a red X.

Appendix

Sitemap

The following table provides an overview of the Event Manager Web service structure.

Interfaces				
	Interface SIP-DECT			
		General		
		Endpoints		
		User defined event text		
			Text replacement	
			Event text structure	
		Import endpoints		
			Endpoints assigned	
			Endpoints available	
	Interface ESPA			
		General		
		Endpoints		
		User defined event text		
			Text replacement	
			Event text structure	
		Event assignment		
		Simulator/Trace		
			Simulator	
			Trace	
	Interface SNMP			
		General		
		Endpoints		
		Event assignment		
		Simulator/Trace		
			Simulator	
			Trace	
	Interface Modbus	General		
		Endpoints		
			Endpoint config	
		Simulator/Trace		
			Inputs	
			Outputs	
	Interface MQTT			
		General		

		Endpoints		
		User defined event text		
			Text replacement	
			Event text structure	
		Topics		
		Subscribe mapping		
		Publish mapping		
	Interface Web-API			
		General		
		Endpoints		
	Interface Web-Event			
		Endpoints		
		Web events		
	Interface IP-Phone			
		General		
		Endpoints		
		URIs		
		Trace		
	Interface GPS			
		General		
		Kmz files		
Event types				
Notification profiles				
	SIP-DECT profile			
	IP-Phone profile			
Notification groups				
	Label			
		Endpoints assigned		
		Endpoints available		
	Description			
	Address			
Event plans				
	Plan			
		Filter: Event type		
			Event types assigned	
			Event types available	
		Filter: Location		
			Locations assigned	

			Locations available	
		Filter: Timetable		
			Start time (hh:mm)	
			End time (hh:mm)	
			Days of the week	
		Phase		
			Endpoints	
				Endpoints assigned
				Endpoints available
			Notification groups	
				Notification groups assigned
				Notification groups available
			Settings	
				Duration (sec)
				Number of retries
				Number of confirmations
				No notification to originating endpoint
				Callback address
		Settings		
			Restart plan after completion	
			Continue running plan on same event	
Locations				
	Location			
		Endpoints assigned		
		Endpoints available		
Users				
	Name			
	Permission			
	Password			
	Password confirmation			
System				
	General			
	Backup/Restart			

	Security			
	Security level			
		Security level		
		Cipher suites		
		Use defaults (switch)		
		Used cipher suites		
		Supported cipher suites		
	Console			
	CloudLink			
Overview				
	All			
	Event flow	Endpoints filter	Event types filter	
	Plan execution flow			
	Notification groups			
	Interface endpoint relations			
	MQTT mappings			
Monitor				

Web UI Parameter, Action & Status Information overview

Event Manager without Locating

Web UI Parameter, Action & Status Information		Description
Interfaces	Configuration pane to administrate the Event Manager's interfaces. Up to 10 interfaces are supported. There is always one SIP-DECT interface which cannot be deleted.	
	Active	Switch to activate or deactivate the interface
	State	Shows the state of the interface (running, misconfigured, inactive)
	Label	Name to identify the interface
	Description	Additional information
	Type	SIP-DECT, ESPA, SNMP, MODBUS, MQTT, Web-Api, IP-Phone, Web-Event, GPS
	Endpoints	Shows the number of configured endpoints for the interface. Up to 2000 endpoints in total are supported across all interfaces.
Type SIP-DECT	There is one interface to connect with the SIP-DECT OMM. Standby-OMM configuration is supported. Via this interface, messages are sent to SIP-DECT telephones, confirmations as well as alarm triggers are received from telephones, e.g., SOS, Man Down or Alarm Trigger.	
	General	General settings for the SIP-DECT interface
	OMM 1	OMM IP address
	OMM 2	Standby OMM IP address
	User	Username to authenticate with the OMM
	Password	Password to authenticate with the OMM
	User defined event text	Switch to activate or deactivate the user defined event text function
	Endpoints	Via SIP-DECT reachable endpoints (SIP-DECT users)
	Active	Switch to activate or deactivate the endpoint
	Address	Endpoint identifier e.g., telephone number
	Label	Endpoint name
	Location	Location to which the endpoint is assigned
	User defined event text	The user defined event text feature allows to modify or replace the received event text to generate an appropriate notification.
	Text replacement	Simple text replacement function. Up to 10 text replacement rules can be defined.
	Text	Text to be replaced
	Replace by	Replacing text

Web UI Parameter, Action & Status Information		Description
	Event text structure	Function to create a new text from predefined elements. The user defined event text can be composed of up to 4 elements.
	Text	One of the following elements: Event type, Event type short, Priority, Originating endpoint (name), Originating endpoint (address), Location of originating endpoint, Event phase, Received text from interface
	Max. length	Maximum length of text to be inserted
	Spacer	Separator to separate the text elements
	Import endpoints	Function to simplify the setup of SIP-DECT endpoints
	Endpoints assigned	Endpoints which are already imported from SIP-DECT into EVP
	Endpoints available	SIP-DECT endpoints that can still be imported
Type ESPA	Incoming Interface to connect with a nurse call system, fire alarm system or similar via ESPA 4.4.4 over IP.	
	General	General settings for the ESPA interface.
	IP address	IP address of the nurse call system or similar or of the serial IP converter to connect with
	IP port	IP port of the nurse call system or similar or of the serial IP converter to connect with
	Interface supervision	Switch to enable or disable interface monitoring
	Determine endpoint by	Switch for defining the method for determining the endpoint. One of the two options: Call address, Message text
	Default event type	Event type that should be used if no other event type was determined
	Call type 1 (Field 4) terminates event	Switch to activate or deactivate the option that Call type 1 (ESPA Field 4) shall terminate the event
	User defined event text	Switch to activate or deactivate the user defined event text function
	Endpoints	Endpoints that can send events to the Event Manager via the ESPA interface.
	Active	Switch to activate or deactivate the endpoint
	Address	Endpoint identifier e.g., ESPA call address
	Label	Name to identify the endpoint
	Location	Location to which the endpoint is assigned

Web UI Parameter, Action & Status Information		Description
	User defined event text	The user defined event text feature allows to modify or replace the received event text to generate an appropriate notification.
	Text replacement	Simple text replacement function. Up to 10 text replacement rules can be defined (not usable for event type, priority and phase)
	Text	Text to be replaced
	Replace by	Replacing text
	Event text structure	Function to create a new text from predefined elements. The user defined event text can be composed of up to 4 elements.
	Text	One of the following elements: Event type, Event type short, Priority, Originating endpoint (name), Originating endpoint (address), Location of originating endpoint, Phase, Received text from interface
	Max. length	Maximum length of text to be inserted
	Spacer	Separator to separate the text elements
	Event assignment	Function for assigning an event type based on different ESPA 4.4.4 message contents.
	Position	Position of the rule in the list of rules. First matching rule will be applied.
	Ringtone (3)	Ringtone value (ESPA field 3) which should be mapped to the specified event type.
	Priority (6)	Priority value (ESPA field 6) which should be mapped to the specified event type.
	Text (2)	Text value (ESPA field 2) which should be mapped to the specified event type.
	Event type	Event type to be used.
	Text position	Start position in the received event text from which the event text should be copied. 0 - the original event text will be used.
	Text length	Number of characters that should be taken over from the received event text from the start position.
	Event text	Alternative text to replace or add the original event message text.
	Separator	Delimiter which will be followed by a phone number, e.g. for callback
	Simulator/Trace	

Web UI Parameter, Action & Status Information		Description
	Simulator	The simulator function allows to send ESPA messages into the Event Manager to emulate traffic even when the interface is not connected to another system.
	Call address	ESPA Field 1 Call address (mandatory field)
	Display message	ESPA Field 2 Display message (mandatory field)
	Ring tone	ESPA Field 3 Ringtone
	Call type	ESPA Field 4 Call type
	Priority	ESPA Field 6 Priority (1 – alarm, 2 – high, 3 – normal)
	Trace	Function to display traffic on the ESPA interface
	Data received	Switch to enable display of received data
	Data sent	Switch to enable display of sent data
	Vital sign	Switch to enable display of keep alive messages / ESPA polling messages
	View Hex	Switch to enable display of data additionally in hexadecimal format
	Trace window	ESPA traffic display window
Type SNMP	The SNMP interface allows to send SNMPv2c traps or inform-requests to a trap destination and to receive trap messages from SNMP clients.	
	General	General settings of the SNMP interface.
	Notification sending	Switch to activate/deactivate the SNMP sender
	IP address	IP address of the SNMP server (trap receiver).
	IP port	IP port address of the SNMP server (trap receiver) (default: 162).
	Type	Either trap or inform message can be selected.
	Community send	SNMP trap community for SNMP message sending, e.g. ‘public’.
	Notification receiving	Switch to activate/deactivate the SNMP receiver
	Community receive	SNMP trap community for SNMP message receiving, e.g. ‘trapper’.
	IP port listen	IP port address of the SNMP listener (default: 162).
	Endpoints	Endpoints of the SNMP interface.

Web UI Parameter, Action & Status Information		Description
	Active	Switch to activate or deactivate the endpoint
	Address	Endpoint identifier e.g., SNMP receiver call address or SNMP endpoint IP address from which the Event Manager will receive SNMP notifications
	Label	Name to identify the endpoint
	Location	Location to which the endpoint is assigned
	Event assignment	Function for assigning an event type based on different Object identifiers received in SNMP notifications from SNMP endpoints
	Label	Name to identify the endpoint
	Object identifier	MQTT object identifier
	Ignore indices	Number of bytes to ignore in an object identifier
	Event type	Event type to be triggered when the object identifier is received.
	Re-trigger event timeout	Timeout before retriggering the same event due to received object identifier in a SNMP notification
	Units	Short text to be appended to the defined OIDs interpreted data; matches the UNITS clause inside MIB definitions.
	Display hint	Select how defined OIDs value is supposed to be displayed inside the generated notification event text. Values that would lead to useless results are discarded upon event text generating; matches the DISPLAY-HINT clause inside MIB definitions (simplified to a drop-down menu here).
	Simulator/Trace	
	Simulator	The simulator function allows to send SNMP messages into the Event Manager or to receive SNMP messages to emulate traffic even when the interface is not connected to another system.
	Send	
		Type: Coldstart, Event (Man Down) or Status change (current)
	Receive	
		Endpoint IP address

Web UI Parameter, Action & Status Information		Description
		SysUpTime (cs)
		TrapOID
		OID
		Value
	Trace	Function to display traffic on the SNMP interface
	Data received	Switch to enable display of received data
	Data sent	Switch to enable display of sent data
	Additional info	Switch to enable display of data additionally in hexadecimal format
	Trace window	SNMP traffic display window
Type Modbus	The Modbus interface allows to connect external devices (WAGO/MOXA) with incoming and outgoing ports.	
	General	General settings of the Modbus interface.
	IP address	IP address of the Modbus device.
	IP port	IP port address of Modbus device.
	Endpoints	Endpoints of the Modbus interface.
	Active	Switch to activate or deactivate the endpoint
	Outgoing	Endpoints to which the Event Manager can send messages
	Incoming	Endpoints from which the Event Manager can receive messages
	Event type	Event type to be used
	Idle current	Switch to activate or deactivate idle current for this endpoint
	Alarm delay (sec)	How long the endpoint needs to be activated in order to trigger an event in seconds
	Behavior when returning to normal state	Select the behavior of this endpoint when it returns to its normal state (e.g. “Do not terminate event”, “Terminate event immediately” & “Terminate event at the end of phase”)
	Address	Endpoint identifier e.g., MODBUS call address
	Label	Name to identify the endpoint
	Location	Location to which the endpoint is assigned

Web UI Parameter, Action & Status Information		Description
	Simulator/Trace	
	Trace	The trace window shows if connection to a Modbus device could be established or not (errors) and if it is possible to received trigger events from incoming endpoints.
	Simulator	The simulator function allows simulation of events on incoming endpoints into the Event Manager to emulate traffic even when the interface is not connected to anything. The status of incoming and outgoing endpoints from a real connected Modbus device will also be shown.
Type MQTT	Interface to connect with a MQTT broker via MQTT protocol.	
	General	General settings for the MQTT interface.
	IP address	IP address of the MQTT broker to connect with
	IP port	IP port of the MQTT broker to connect with
	User defined event text	Switch to activate or deactivate the user defined event text function
	User	User name for authentication
	Password	Password for authentication
	Use TLS	Switch for activation of TLS protocol for the communication
	Validate certificates	Switch for activation of certificate validation (used with TLS protocol)
	Endpoints	IoT devices from which the Event Manager can receive events via the MQTT interface to the MQTT broker.
	Active	Switch to activate or deactivate the endpoint
	Address	Endpoint identifier e.g., identifier of the IoT device publishing events to the same MQTT broker
	Label	Name to identify the endpoint
	Location	Location to which the endpoint is assigned
	User defined event text	The user defined event text feature allows to modify or replace the received event text to generate an appropriate notification.
	Text replacement	Simple text replacement function. Up to 10 text replacement rules can be defined (not usable for event type, priority and phase)

Web UI Parameter, Action & Status Information		Description
	Text	Text to be replaced
	Replace by	Replacing text
	Event text structure	Function to create a new text from predefined elements. The user defined event text can be composed of up to 4 elements.
	Text	One of the following elements: Event type, Event type short, Priority, Originating endpoint (name), Originating endpoint (address), Location of originating endpoint, Phase, Received text from interface
	Max. length	Maximum length of text to be inserted
	Spacer	Separator to separate the text elements
	Topics	MQTT topic for subscription or publishing
	Active	Switch to activate or deactivate the topic
	Type	Type of topic (Subscribe or Publish)
	Message as payload	Switch to select if notification message should be sent as payload or not in a publish message to the MQTT broker
	Endpoint	Label of the endpoint to which the topic is assigned to
	Subscribe mapping	MQTT topic for subscription or publishing
	Active	Switch to activate or deactivate the subscribe mapping
	Event type	Type of event which shall be triggered by the received MQTT message
	Condition	Condition which will be checked to trigger an event when a MQTT message for a subscribed topic is received (Same text, Contain text, Value equal, Value greater, Value smaller)
	Publish mapping	Mapping of an event type to a publish topic with payload content
	Topic	Publish topic to be send to the IoT device via the MQTT broker
	Event type	Type of event that will trigger the publish message to MQTT message
Type Web-API	Interface to deal with a Web application via HTTPS protocol (RESTapi).	
	General	General settings for the Web-API interface.
	Incoming URL	Fix: https://<IP address of the EM> or <CLD tunnel> /wapi/v1/request
	URL: event	Incoming URL

Web UI Parameter, Action & Status Information		Description
	URL: event result	URL for outgoing responses to requested events
	URL: event cancel	Incoming URL
	URL: notification	URL for sending of outgoing notifications
	URL: confirmation	Incoming URL
	URL: cancel	URL for canceling of outgoing notifications
	API key	Buttons for 'Copy to clipboard' and 'Renew' of API key (CloudLink-API)
	Validate certificates	Switch to enable certificate validation during outgoing messages
	Endpoints	Internal endpoints for sending/receiving Web-API notifications/requests.
	Active	Switch to activate or deactivate the endpoint
	Address	Endpoint identifier e.g., identifier of the Web-API device requesting events or receiving notifications
	Label	Name to identify the endpoint
	Location	Location to which the endpoint is assigned
Type Web-Event	Interface to deal with events triggered from web admin	
	Endpoints	Internal endpoints for sending/receiving Web-API notifications/requests.
	Active	Switch to activate or deactivate the endpoint
	User	Web admin user with ability to trigger web events from a list
	Web events	List of configured web events
	Label	Name to identify the type of web event
	Event type	Event type to be triggered
	Event plan	Event plan to be executed
	Text	Text to be notified (can be changed during the triggering process)
Type IP-Phone	Interface to deal with IP-Phones (Mitel-SIP and MINET)	
	General	General settings for the IP-Phone interface.
	Validate certificates	Switch to activate or deactivate the certificates validation
	Endpoints	Internal endpoints for sending/receiving Web-API notifications/requests.
	Active	Switch to activate or deactivate the endpoint

Web UI Parameter, Action & Status Information		Description
	Address	SIP username received from the polling IP-Phone
	Label	Identifier for the endpoint (e.g. name of the phone's user in the PBX)
	Location	Location to which the endpoint is assigned
	URIs	List of URIs to be configured on the IP phone
	Poll	URI for polling the Event Manager by the phone
	Event	URI for triggering an event in the Event Manager
	Poll 2	URI for polling the Event Manager by the phone (EM redundancy)
	Event 2	URI for triggering an event in the Event Manager (EM redundancy)
	Config file for MiVoice Business	Download links for configuration files of different phone types
	Trace	Function to display traffic on the IP-Phone interface
	Data received	Switch to enable display of received data
	Data sent	Switch to enable display of sent data
	Additional info	Switch to enable additional information
	Trace window	IP-Phone traffic display window
Type GPS	Interface to deal with GPS data server(s)	
	General	General settings for the GPS interface.
	IP-Adresse	IP address of the first GPS data server
	IP Port	IP port of the first GPS data server
	IP-Adresse 2	IP address of the second GPS data server
	IP Port 2	IP port of the second GPS data server
	XML-ID	XML-ID of the GPS XML application as configured in the OMM
	Warning if other Regulatory Domain is needed	Time in hours before the needed switching of the DECT Regulatory Domain
	Default Regulatory Domain	Setting of the default DECT Regulatory Domain
	Kmz files	List of loaded KMZ files (with polygons for DECT regularity domains)

Web UI Parameter, Action & Status Information		Description
Event types	Configuration pane to administrate up to 100 event types. Individual events are mapped to these event types for further processing.	
	Label	Event type name
	Short text	Short (max. 8 character long) event type name
	Priority	Event priority
	Description	Additional information
Notification profiles	Configuration pane to administrate up to 50 notification profiles. Notification profiles define the way notifications are presented by the receiving device.	
	Label	Notification profile name
	Description	Additional information
	SIP-DECT profile	The profile contains various parameter to control the way a notification is indicated on the Mitel 6x2d/700d DECT phone.
	Ringtone group	The Event Manager can control the ringtone to alert the message received on the DECT phone. Various options are available: a) Not to be used for now: None b) Using the device settings with selection of a specific melody setting: Local settings c) Selecting a ringtone from a group: one of the available ringtone groups
	Ringtone	a) If the ringtone group is set to “Local settings”, a specific melody setting of the device can be selected. B) If a ringtone group is set, a melody or sound effect can be selected.
	Priority	SIP-DECT message priority: Low, Normal, High, Emergency
	Ring volume	Ring tone volume which shall be used to indicate the notification.
	Increasing ring volume	Enables the automatic volume increase
	Vibration	Enables the vibration function if not automatically activated by the phone based on the message priority.
	No alert tone during call	Switch to turn off the audible indication (in-band) of the received message.

Web UI Parameter, Action & Status Information		Description
	Message logging	Switch to turn on the message logging on the phone for accepted and rejected messages.
	Disconnect existing call	If activated, ends an existing telephone conversation when the message arrives.
	Font color	Color of the message text
	Background color	Color of the background
	IP-Phone profile	The profile contains various parameter to control the way a notification is indicated on the IP-Phone.
	Ringtone	Selection of ring melodies (Alarm 1 .. Alarm 7 or Ringtone off)
	Ring volume	Volume for ringing during notifications (1 .. 10)
	Call protection	Setting for Call protection (No, Yes, Yes with Info)
	Beep	Switch to enable a message beep (important for MINET IP-Phones)
	Font color	Color of the message text
	Background color	Color of the background
Notification groups	Configuration pane to administrate up to 50 notification groups. (maximum 2000 endpoints in total across all groups). Notification groups group endpoints to be notified for easier management. Groups can be assigned to phases of event plans instead of individual endpoints. In addition, notification groups can have addresses to use the "Use call address" function in event plans.	
	Label	Notification group name
	Description	Additional information
	Address	Unique id e.g., telephone number / extension number
	Endpoints assigned	List of endpoints assigned to this group (Label/Address)
	Endpoints available	List of endpoints which could be assigned to this group (Label/Address)
Event plans	Configuration pane to administrate up to 500 event plans. Event plans define processes for handling received events sent by endpoints at the various locations to notify receiving endpoints	
	Active	Switch to activate or deactivate the event plan.
	Label	Event plan name
	Description	Additional information
	Filter: Event type	

Web UI Parameter, Action & Status Information		Description
	Event types assigned	List of Event types for which the plan is applied, i.e., should be executed.
	Event types available	List of Event types that have not yet been assigned to the plan, i.e., to which the plan is not applied
	Filter: Location	
	Locations assigned	List of Locations to which the plan applies, i.e., the plan is applied to events sent from endpoints assigned to these locations.
	Locations available	List of Locations that have not yet been assigned to the plan, i.e., to which the plan does not apply
	Filter: Timetable	
	Start time (hh:mm)	Start time for the validity of the plan
	End time (hh:mm)	End time for the validity of the plan
	Days of the week	Checkboxes for the validity of the plan on the days of a week
	Phase	Event plan phases: up to 10 phases in a single plan and up to 1000 phases in total across all event plans.
	Label	Phase name
	Description	Additional description for the phase.
	Use call address	Option to enable selecting a notification group based on the receiving endpoints address. A notification group with the same address must exist.
	With Notification profile	If the notification group is selected by the endpoints call address, then the specified notification profile will be applied when processing this phase.
	Endpoints	Tab in which to be notified endpoints are assigned to the phase.
	Endpoints assigned	Endpoints assigned to this phase (Label / Address).
	Endpoints available	Endpoints which could assigned to this phase.
	Notification profile	Notification profile to be used in this phase for the selected assigned endpoint
	Notification groups	Tab in which to be notified notification groups are assigned to the phase.

Web UI Parameter, Action & Status Information		Description
	Notification groups assigned	Notification groups assigned to this phase (Label/Address).
	Notification groups available	Notification group which could assigned to this phase (Label/Address).
	Notification profile	Notification profile to be used in this phase for the selected assigned group
	Settings	Tab for configuring general phase settings.
	Duration	Duration in seconds
	Number of retries	Never / Permanently / 1..49
	Number of confirmations	Individual (each endpoint) or value between 1 and 49
	No notifications to originating endpoint	Switch to disable notifications to originating endpoint(s)
	Callback address (if not provided yet)	Callback address for dialing at notification endpoint (DECT phone)
	Settings	Special settings of the plan
	Restart plan after completion	Switch to enable or disable the restart of the plan after completion (default: off)
	Continue running plan on same event	Switch to enable or disable the continuation of the plan by the same event (default: off)
Locations	Configuration pane to administrate up to 500 endpoint locations. These locations can be assigned here endpoints that probably will send events to the Event Manager. The locations can be used as filter in Event plans so that location-dependent processes can be defined. If different locations are configured, they are displayed in the menu as an expandable tree.	
	Location	Complete location information with parent locations
	Label	Location name
	Description	Additional information
	Endpoints assigned	List of endpoints assigned to this location (Label/Address).
	Endpoints available	List of endpoints which are not assigned to any location and could assigned to this location (Label/Address).

Web UI Parameter, Action & Status Information		Description
Users	Configuration pane to administrate up to 10 users who have access to the Event Manager's Web service.	
	Name	Username, login name
	Permission	Permission of the user (Configuration, Monitor, Locating)
	Password	User password
	Password confirmation	User password confirmation
System	Administration pane for various administrative activities for the operation of the Event Manager.	
	General	General system settings
	System name	System name
	CloudLink enabled	Switch to enable or disable the CloudLink daemon
	CloudLink status	Displays the status of the CloudLink daemon
	Version	Running software version is shown here
	Redundancy configured	Displays the status of Redundancy configuration
	Redundancy connected	Displays the status of Redundancy connection
	Watchdog	Switch to enable or disable the Watchdog functionality
	Watchdog IP address	IP address of the watchdog that is to be triggered
	Backup/Restart	Options to restart the Event Manager, backup the configuration and the event log.
	Restart	Restart the Event Manager
	Restart with factory defaults	Restart the Event Manager and resets the Event Manager configuration to default
	Export log	Allows to store the event logs on the PC as a summary and a details logfile (format .csv)
	Export config	Allows to store the Event Manager's configuration on the PC as a <system name>_<date>_<time>_em_conf.gz file
	Import config	Allows to restore the Event Manager's configuration from a PC
	Security	Options to import trusted certificate, local certificate chain and private key (with or without password).
	Trusted certificate(s)	Displays how many certificates the Event Manager has

Web UI Parameter, Action & Status Information		Description
	Local certificate chain	Displays how many local certificate chains the Event Manager has
	Private key	Display if the Event Manager has a working private key
	Private key: password	Enter the password for the imported private key
	Private key: password confirmation	Confirm the password for the imported private key
	Import PEM file with	Define the type of PEM file to be imported
	Import PEM file	Import a PEM file
	Delete certificates/key	Delete all imported certificates and keys
	Make it work	Button for restart of the Event Manager to apply changes
	File name	List of loaded .pem files with trusted certificates
	Trusted certificate(s)	Number of trusted certificate(s) included in the .pem file
	Security level	Options to configure the security level used by the Event Manager and the used cipher suites for AXI and HTTPS connections.
	Security Level	Select the security level (“Legacy”, “Medium” or “High”)
	Cipher suites of security level	Select the cipher suites security level (“Legacy”, “Medium” or “High”)
	Use defaults	Switch to use or not use the default settings
	Used cipher suites	List of all used cipher suites (can be edited)
	Supported cipher suites	List of all supported cipher suites
	Console	Access to system console without access to the root shell
	CloudLink	Shows the current configuration of the CloudLink Daemon and allows to configure connection to CloudLink portal and for Remote Management.
Overview	Area to display the currently configured event flow, notification groups, interface endpoint relations and MQTT mappings	
	Event flow	Table with Endpoint/Interface, Events, Locations, Plans and Timetables, probably filtered by Endpoint and/or Event types
	Plan execution flow	Table with Plan, Phase, Notification endpoint/group and Profile
	Notification groups	Table with Group, Call address and Notification endpoint
	Interfaces endpoint relations	Table with Interfaces and their connected endpoints

Web UI Parameter, Action & Status Information		Description
	MQTT mappings	Table with Endpoints, Topics, Conditions, Events and Publish payload
Monitor	Area to display the currently active event processing activities and their status and an option to terminate them.	
	Cancel all	Cancel all active event plans
	Export log	Allows to store the event logs on the PC as a summary and a details logfile (format .csv)
	Priority	Event type priority
	Type	Event type
	Text	Event message text
	Endpoint	Endpoint that triggered the event
	Phase	Current event plan phase
	Confirmations	Received Confirmations/Required Confirmations
	Cancel	Cancel a single active event plan

Event Manager with Locating

In case of an Event Manager running as PC application on a Linux server and connected to an OMM with an installed *'Mitel SIP-DECT Locating Server License'* an additional menu entry in the menu tree is available: Locating.

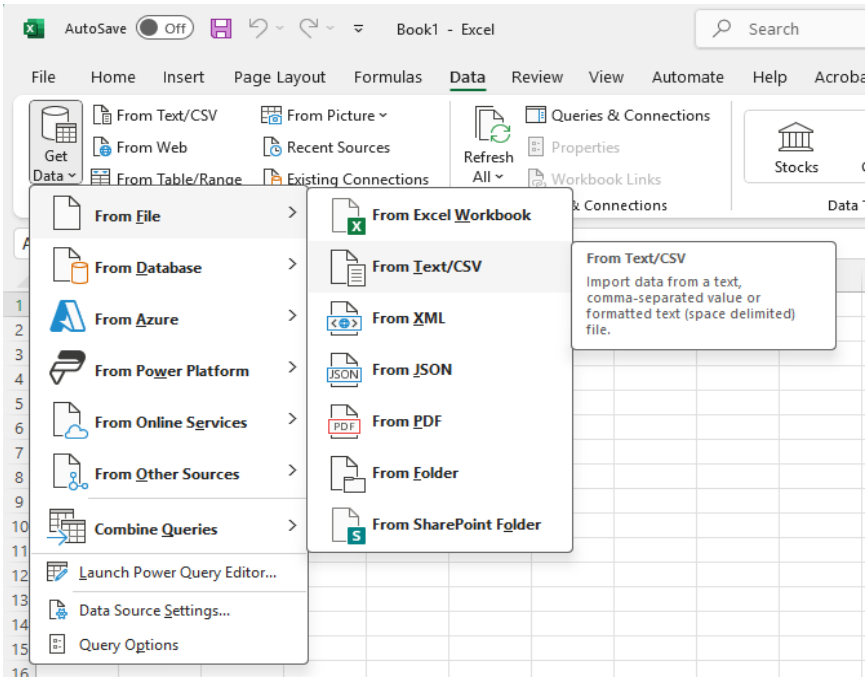
This menu entry is realized as an expandable tree with locations configured in the Event Manager and includes different tabs for Monitor, Users, Maps and Radio Fixed Parts (RFPs) and Beacons. Here is also available a button 'Import locations' to import locations that are already defined in the OMM.

Web UI Parameter, Action & Status Information		Description
Locating	Configuration pane realized as an expandable tree with locations configured in the Event Manager	
	Monitor	Area to display the currently active event processing activities and their status and an option to terminate them.
	Cancel all	Cancel all active event plans
	Export log	Allows to store the event logs on the PC as a summary and a details logfile (format .csv)
	Trigger event	Trigger an event via Web event interface (only visible for configured ujsers)
	Priority	Event type priority
	Type	Event type
	Text	Event message text
	Endpoint	Event type priority
	Phase	Current event plan phase
	Confirmations	Received Confirmations/Required Confirmations
	Cancel	Cancel a single active event plans
	Users	List of SIP-DECT users with activated locatable and/or trackable feature
	Name	Username assigned to a locatable DECT device (in SIP-DECT)
	Phone number	Phone number of the DECT device (in SIP-DECT)
	Location	Actual location of the DECT device (based on events from OMM)
		Link to a map showing the current location of the DECT device
	On	Icon to show if the DECT device position has been received already before
	Last action	Date and time of the last known position of a DECT device (based on events from the OMM)

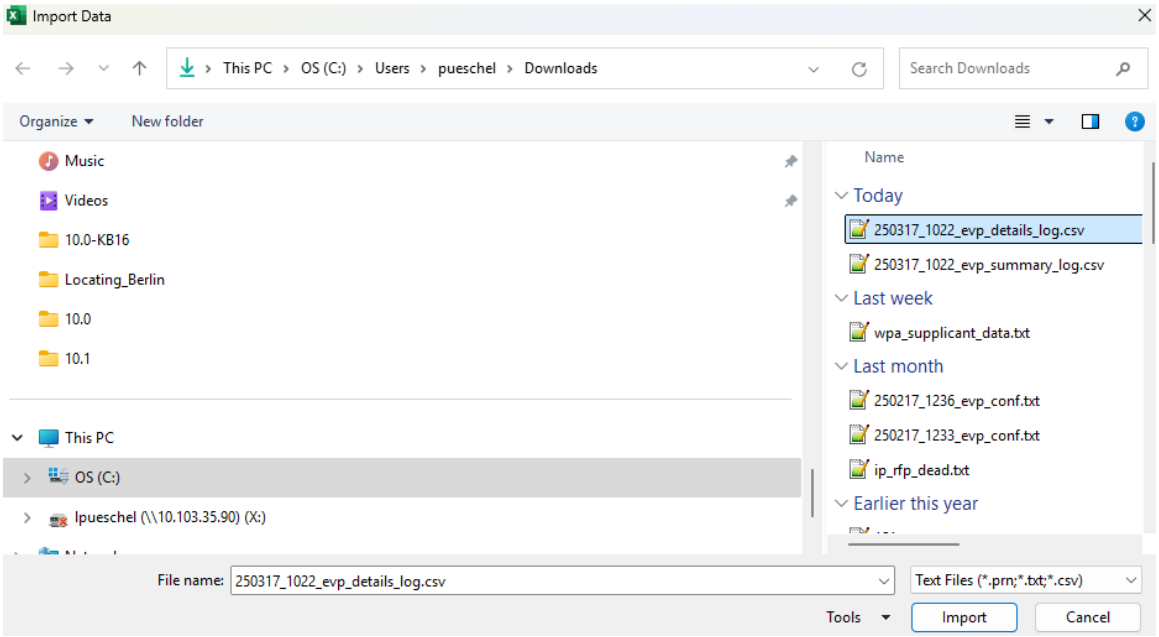
Web UI Parameter, Action & Status Information		Description
	Description 1	Username assigned to a locatable DECT device (in SIP-DECT)
	Description 2	Cancel a single active event plan
	Maps	List of loaded maps
	Label	Description for the loaded map
	Image	Link to the loaded map
	Zoom level	Available zoom levels of a loaded map
	Location	Username assigned to a locatable DECT device (in SIP-DECT)
	Save / Delete	Buttons for Saving / Deleting an entry
	RFPs	List of SIP-DECT Radio Fixed Parts (RFP) (automatically imported from the OMM)
	Name	Username assigned to a locatable DECT device (in SIP-DECT)
	MAC address	MAC address of the RFP
	Location	Assigned location of the RFP
	Detail	Icon that shows if the RFP is already positioned on a loaded detail map
	Overview	Icon that shows if the RFP is already positioned on a loaded overview map
	Beacons	List of Bluetooth beacons (probably imported from Excel file)
	Name	Name of Beacon
	Description	Description for the location (e.g. floor, room, etc.)
	Major	Major number of the beacon
	Minor	Minor number of the beacon
	Detail	Icon that shows if the beacon is already positioned on a loaded detail map
	Overview	Icon that shows if the beacon position is already available on a loaded overview map

Recommendation on the procedure for importing log data into Microsoft Excel

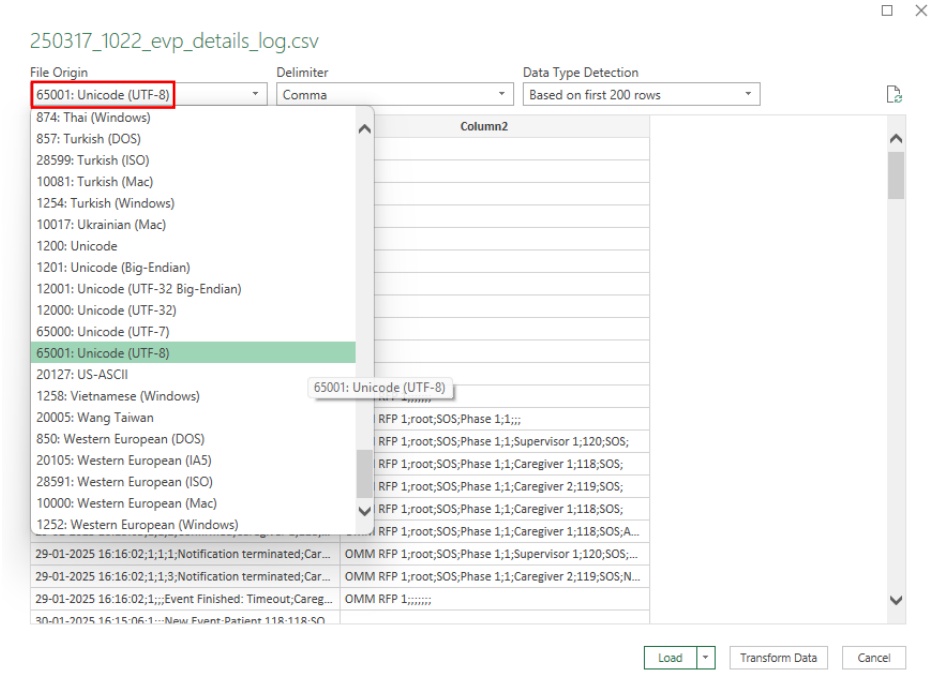
A convenient way to import the log data from the Event Manager into a Microsoft Excel file while maintaining the correct formatting of the data is the data import ‘Get data from file’.



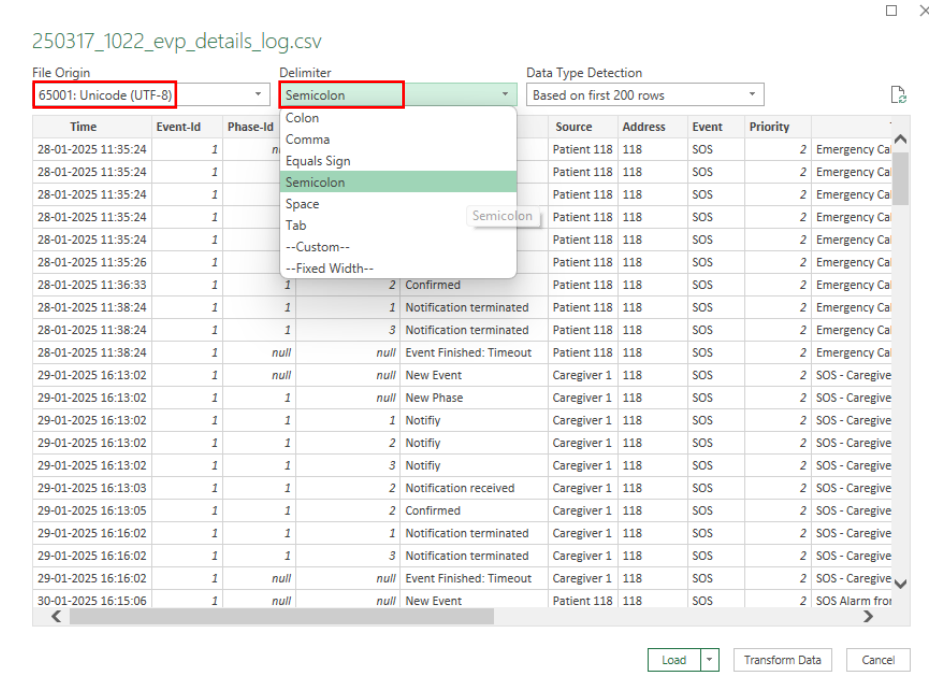
Select the file you want to import.



Select “Unicode (UTF-8)” for the encoding so that non-ASCII characters are displayed correctly.

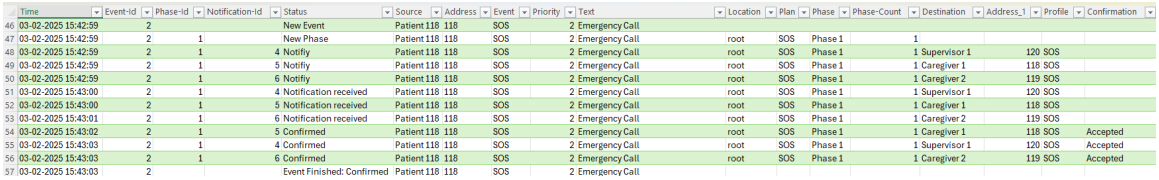


Select “Semicolon” as delimiter.



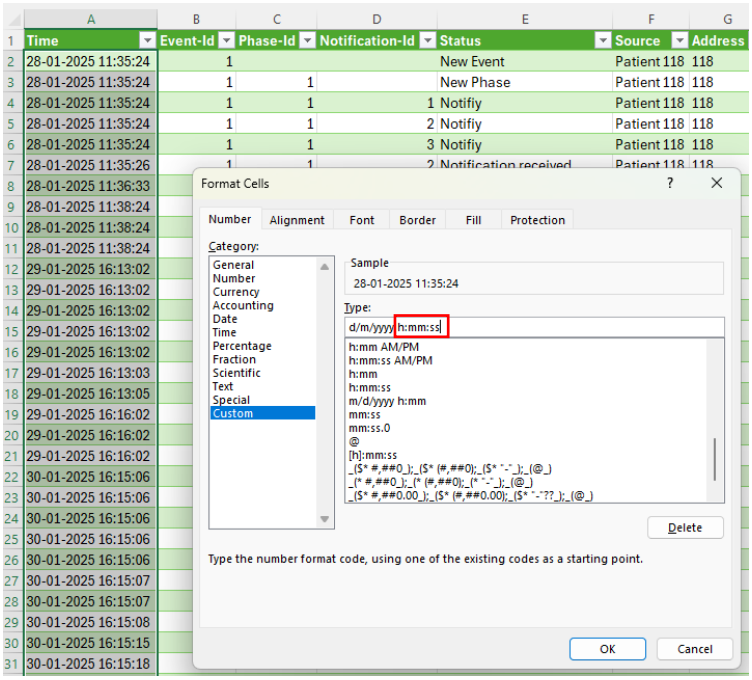
Then confirm with “Load” to load the data.

The data should then be displayed in this way.

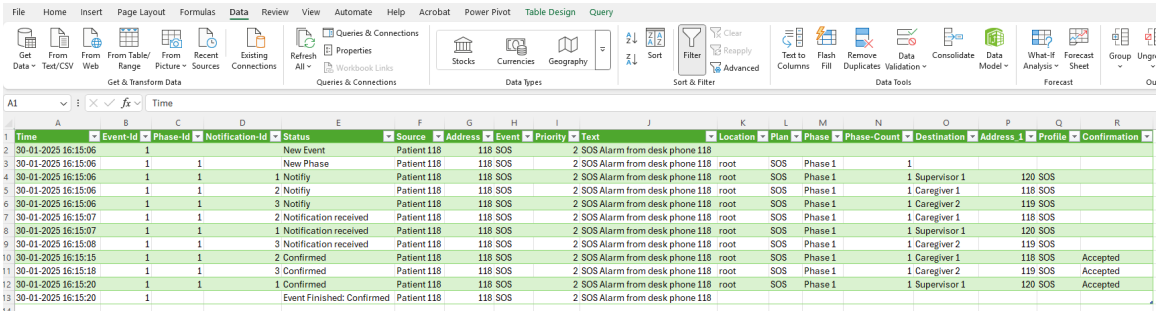


Time	Event-Id	Phase-Id	Notification-Id	Status	Source	Address	Event	Priority	Text	Location	Plan	Phase	Phase-Count	Destination	Address_1	Profile	Confirmation
03-02-2025 15:42:59	2			New Event	Patient 118	118	SOS	2	Emergency Call								
03-02-2025 15:42:59	2	1		New Phase	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1				
03-02-2025 15:42:59	2	1	4	Notify	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
03-02-2025 15:42:59	2	1	5	Notify	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
03-02-2025 15:42:59	2	1	6	Notify	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	
03-02-2025 15:43:00	2	1	4	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
03-02-2025 15:43:00	2	1	5	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
03-02-2025 15:43:01	2	1	6	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	
03-02-2025 15:43:02	2	1	5	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	Accepted
03-02-2025 15:43:03	2	1	4	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	Accepted
03-02-2025 15:43:03	2	1	6	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	Accepted
03-02-2025 15:43:03	2			Event Finished: Confirmed	Patient 118	118	SOS	2	Emergency Call								

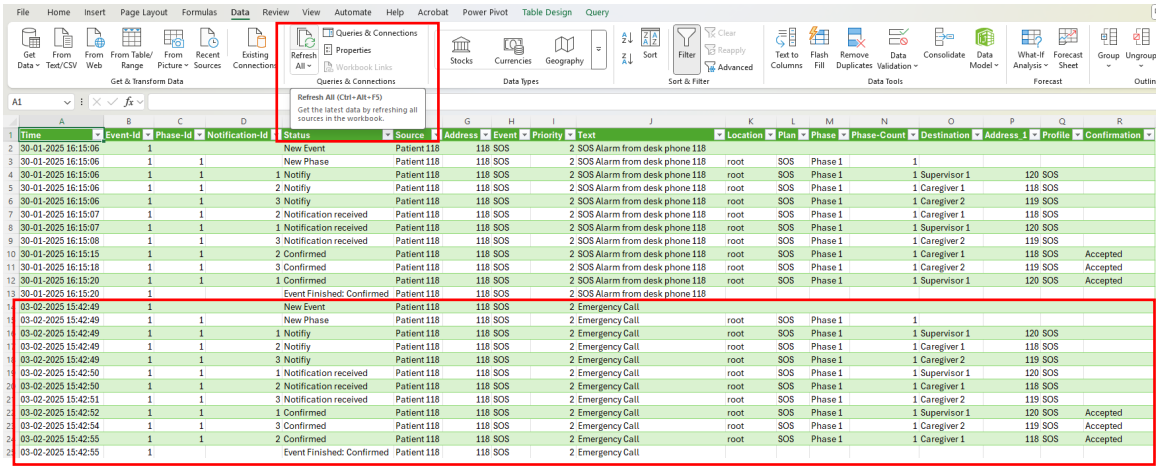
If the time does still not contain seconds, the format of the cells must be adjusted. To do this, select the user-defined format “d/m/yyyy hh:mm” and add “:ss” so that the time consists of hours:minutes:seconds (“d/m/yyyy hh:mm:ss”)



Since the data is linked to the source file, the above steps do not have to be repeated each time. If updated logs are copied to the same location under the same file name, a refresh of the data is sufficient.



The changed data appears after the refresh.



Time	Event-Id	Phase-Id	Notification-Id	Status	Source	Address	Event	Priority	Text	Location	Plan	Phase	Phase-Count	Destination	Address_1	Profile	Confirmation
30-01-2025 16:15:06	1			New Event	Patient 118	118	SOS	2	SOS Alarm from desk phone 118								
30-01-2025 16:15:06	1	1		New Phase	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1			
30-01-2025 16:15:06	1	1	1	Notifiy	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Supervisor 1	120	SOS	
30-01-2025 16:15:06	1	1	2	Notifiy	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 1	118	SOS	
30-01-2025 16:15:06	1	1	3	Notifiy	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 2	119	SOS	
30-01-2025 16:15:07	1	1	2	Notification received	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 1	118	SOS	
30-01-2025 16:15:07	1	1	1	Notification received	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Supervisor 1	120	SOS	
30-01-2025 16:15:08	1	1	3	Notification received	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 2	119	SOS	
30-01-2025 16:15:15	1	1	2	Confirmed	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 1	118	SOS	Accepted
30-01-2025 16:15:18	1	1	3	Confirmed	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 2	119	SOS	Accepted
30-01-2025 16:15:20	1	1	1	Confirmed	Patient 118	118	SOS	2	SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Supervisor 1	120	SOS	Accepted
30-01-2025 16:15:20	1			Event Finished: Confirmed	Patient 118	118	SOS	2	SOS Alarm from desk phone 118								
03-02-2025 15:42:49	1			New Event	Patient 118	118	SOS	2	Emergency Call								
03-02-2025 15:42:49	1	1		New Phase	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1			
03-02-2025 15:42:49	1	1	1	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Supervisor 1	120	SOS	
03-02-2025 15:42:49	1	1	2	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Caregiver 1	118	SOS	
03-02-2025 15:42:49	1	1	3	Notifiy	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Caregiver 2	119	SOS	
03-02-2025 15:42:50	1	1	1	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Supervisor 1	120	SOS	
03-02-2025 15:42:50	1	1	2	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Caregiver 1	118	SOS	
03-02-2025 15:42:51	1	1	3	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Caregiver 2	119	SOS	
03-02-2025 15:42:52	1	1	1	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Supervisor 1	120	SOS	Accepted
03-02-2025 15:42:54	1	1	3	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Caregiver 2	119	SOS	Accepted
03-02-2025 15:42:55	1	1	2	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1		1 Caregiver 1	118	SOS	Accepted
03-02-2025 15:42:55	1			Event Finished: Confirmed	Patient 118	118	SOS	2	Emergency Call								