

OFELIA

Technical Documentation

Version 1.2



Table of Content

Ofelia®: A versatile and innovative solution	14
History	14
Functionalities – Application Fields	14
Events at the heart of the system	15
Notion of Requisition.....	15
Notion of Event.....	15
Natures of events	15
Types of events.....	16
Hierarchy and inheritance	16
Properties	17
Life cycle.....	18
Creation	18
Finite state machine.....	18
System Events.....	18
Technical Alarms.....	18
Nurse calls	19
Missions.....	19
Runaways.....	20
LWP alarms (Lone Worker Protection Alarm)	20
Notifications/Mobilizations	20
Mobilization Campaigns	21
Global state priority.....	21
Connections to the information system.....	22
Drivers and driver instances	22
Capabilities	22
Triggers and Scenarios	26
Workflow engine	26
Definition of workflows.....	26
State machine.....	27
Workflow activities.....	27
Start	27

Call	27
Calling while a zone is occupied	28
Waiting for all calls	28
Wait for a specific time.....	28
Empty the list of users	28
Variable writing	28
Delete messages.....	28
Success, Failure.....	29
Workflow conditions	30
Condition on number	30
Condition on minimum number	30
Condition based on a script.....	30
Requisition rules.....	30
Requisition Mode	31
Requisition filter	32
List Restriction	32
Geographical selection context	32
Triggering rules engine	32
Trigger types.....	32
Trigger conditions.....	33
Order of precedence	33
Specific criteria	33
Workflow parameters reported in the rule.....	34
User management	35
Patients.....	35
Customers.....	35
Employees / Staff members	35
Department / Units	35
Presence Type	36
Functions	36
Planning.....	36
Creation of a schedule.....	36
Absences and on-call duty.....	37
Example	37
Association of the employee with the schedule	37

Availability	38
Login ID	39
Communications and calls	41
Media	41
Media Types	41
Addresses	41
Call profile	41
Call Strategy	42
Media templates	42
Message format	43
Devices	44
Nurse call boxes	44
Medallions	44
Beacons	44
Geographic locations	45
Zones, zone types and positions	45
Inheritance	45
Floor maps	46
Event monitoring and tracking	47
Event monitors	47
Reminder Monitors	47
Out-Of-Time Monitors	47
Quality-of-service measurement	48
Visit reminders	48
Event Priorities	49
Priority groups	49
Change of priority	49
Actions and Forms	49
Security	51
Licenses	51
Licensing options	51
Connection	52
Permissions	52
Granularity	52

Dynamic permissions.....	52
Roles	53
System permissions	54
Lone Worker Protection	58
Detection	58
Temporal alarms.....	58
Mobilization Campaigns	59
Campaigns vs Mobilizations	59
Campaigns Templates.....	60
Campaigns dashboard	60
Operational follow-up	60
Event Generators	61
Variables	61
Alarm generators.....	61
Precedence of states	62
Mobilization generators	62
Mobilization campaign generators.....	62
Files.....	63
Capability converters.....	63
Nurse call to LWP alarm	63
Text to Nurse Call Converter	63
Text to Variable Converter	64
Statistics	65
Custom analysis	65
Dashboards.....	66
Widgets	67
System Administration	68
Installation	68
Backup and Restore	68
Location of backups.....	68
Version compatibility	68
Scheduled Automatic Backups	69
Manual backups	69
Manual restoration	69

Configuration Import and Export	70
Purge	70
Manual purge	70
Scheduled purge.....	70
Exploitation	71
Filtering.....	71
An event-centric approach.....	71
Timeline.....	71
Dashboard Lists	72
Events in progress	72
Pagination and Filter	73
Quick actions	73
History	73
Detail of an event	74
Department/Unit View.....	74
Floor maps.....	75
Users.....	75
Patients.....	75
Clients.....	76
Employees	76
Deployment and Topology	76
Mono Server	77
Multi Servers	77
Supporting multiple tenants.....	77
Extensibility and interfacing	78
User Manual	79
General Guidelines	79
Administrator Guide	80
User interface common concepts	80
Initial setup.....	82
Communications.....	83
Drivers	83
Message Types	84
Call Profiles.....	84

Capability Converters	85
Nurse Call to LWP Alarm	85
Text to Nurse Call	85
Text To Variable.....	85
Blockly® Programming.....	86
Third-Party Applications.....	86
WebHooks	86
Facility	88
Zones	88
New zone creation.....	89
Adding a Beacon.....	90
Adding a nurse call device	91
Adding an alarm generator.....	92
Floor maps managements	93
Vector Files Format	93
SVG Tag.....	93
Functional Tags.....	94
Definition of zones on the map	94
Localization Beacons	95
Zone Types.....	95
Position Types.....	96
Tags.....	97
Loyalty Programs	97
Mobilization.....	98
Employees	99
Assigning a media to the user	100
Assigning a zone to the user.....	101
Assigning a Unit to the user.....	102
Assigning a Login ID to the user	102
Schedules.....	103
Schedule Creation	103
Assigning a Job to the user.....	105
Assigning the user to a schedule	107
Custom Assignment.....	108
Functions (Jobs).....	109

Departments/Units.....	109
Presence Types.....	110
Availability Levels	111
Availability Profiles	111
User Profiles	113
Files.....	115
Sound Profiles.....	115
Media Templates.....	116
Media Types	117
Medias	118
Roles and permissions.....	119
Workflows	121
Triggering Rules	122
Workflow Variables	125
Order of precedence	125
Call Strategies	125
Events	127
Event Types Edition	128
Event Types Hierarchy	128
Common Properties	129
Message Formats	130
Keywords and variables in messages	131
Visibility for each role	133
Editing Event Monitors.....	135
Quality-of-Service Monitors	135
Reminder Monitors	136
Out-Of-Time.....	138
Event state definitions.....	140
Event priorities	140
Forms (Lists of Actions)	141
Choice Lists for Forms	143
Temporal Alarms Profiles	144
Visit Reminders.....	145
Statistical Analysis	147
Time Ranges	147

Statistics Dashboards	147
Widgets.....	149
Choosing a type of analysis	149
Definition of additional filters	150
Choosing a grouping and aggregation mode.....	150
Graphic layout	152
Devices	153
Nurse Call Devices	153
Medallions.....	154
Event Generators	156
Variables	156
Alarms Generators	156
Mobilization Generators	158
Mobilization Campaign Generators	160
Mobilization Campaigns.....	162
Campaigns Dashboard.....	162
Campaign Dashboard	164
System Administration	167
License management.....	167
Data Management.....	168
Backups.....	168
Target Directory.....	169
Compatibility	169
Scheduled Backups.....	170
Every Day.....	171
Every Week.....	171
Every Month	171
Every Year	171
Retention Policy.....	172
Manual Purge	172
Scheduled Purge.....	172
Configuration Backup	173
Operator's Guide	175
Navigation	175
User Profile.....	176

Availability	176
Display Settings	176
Product Information	178
Navigation	179
Events in progress	180
Dashboard	180
Columns.....	181
Filters	182
Quick Actions.....	183
Event Details.....	184
General View	184
Chronological View.....	185
Geographical View.....	186
Floor map view	187
Displaying events.....	188
Displaying event generators.....	188
Displaying staff members present on zone	188
Department / Unit View	189
Zone Details.....	190
Selecting a zone.....	191
Temporal Alarms	192
Technical Alarms	193
Mobilization Campaigns	194
Dashboard	194
Campaign Tracking	195
Campaign progress monitoring	196
Individual mobilizations tracking.....	197
Campaign dashboard.....	198
Missions.....	199
Missions in progress.....	199
Active Missions.....	199
Filter and export	200
Creating a mission	201
Scheduled Missions	203
Periodicity.....	204

Every Day	204
Every Week	204
Every Month	204
Every Year	204
Human Resources	205
Patients	205
Creating and editing a Patient	205
Stays and Movements	206
Clients / Customers	206
Staff Dashboard	206
Event history	208
History tab	208
Columns	209
Filters	210
Nurse Calls Tab	211
Columns	211
Filters	211
Statistics	213
Custom Analysis	214
Filter	214
Analysis Type	215
Grouping and aggregation	215
Number of occurrences analysis	215
Quality-of-Service Analysis	216
Transition duration analysis	217
APPENDIX I: Drivers' configuration	219
AeroScout Driver - STANLEY Healthcare®	219
ASCII Driver	219
Remote Socket Connection	219
Local Socket Server Connection	220
Serial Port Connection	221
ASCOT IP® Driver	222
ASCOT IP® One Way Driver	222
DIGITAL VIRGO® Driver	223
Email Driver	224

ESPA Driver	225
TCP Connection	225
Serial Link	226
HighSMS Driver.....	226
HTTP Driver.....	228
MiniFinder® Driver	228
MODBUS® Driver.....	228
TCP Connection	229
Serial Link	229
Parameters	230
MultiTech® Driver.....	230
Virtual Nurse Call Driver	231
Ofelia Messenger Driver	231
OfeliaSmart Driver	232
OPC UA Driver.....	232
Opera Driver	233
Ping Driver	234
Pop-up Driver	234
SIA® Driver	234
Main Settings.....	235
Communication	235
Messages Content	236
SIA-DCS Format	236
ADM-CID (Contact Id) Format	236
ADM-CID Custom Format	237
SIP Driver	237
SMS Factor® Driver.....	238
SMTP Server Driver.....	239
Stanley® Driver	239
teleCARE® IP Driver	240
teleCARE® M Driver	241
UDP Sockets.....	241
Serial Link	241
Common Settings	242

APPENDIX II – External Applications..... 243

FILES Application 243

STANLEY® Application 245

TELECARE® IP Application 246

LDAP Directory Application 247

Ofelia®: A versatile and innovative solution

History

Ofelia® is a requisition management application that enables contextual mobilization of employees. Ofelia® solution is a decentralized application for collecting alarms and probes aimed at securing, mobilizing and requisitioning the most relevant staff members for a given alert or mission. Benefiting from 10 years of functional experience on Mercury® software, the Ofelia solution® has been entirely rebuilt on a new technological foundation.

Functionalities – Application Fields

Ofelia® covers the entire cycle of staff member requisitioning and is perfectly adapted to different business sectors, offering a unified solution that meets the specific needs of each business, in particular:

Healthcare: Specific management of nurse calls and the constraints specific to healthcare information systems. In this respect, Ascom benefits from a deep expertise in nurse call solutions, thanks in particular to an exclusive partnership established since its creation with ASCOM France. The concepts of medical departments, room displays management, emergency plan, etc. are integrated into the hospital offer.

Industry: Thanks to its specific connectors, Ofelia® is perfectly integrated with monitoring and follow-up solutions for PLCs (Modbus®, BACnet®, etc.) and also offers a complete range of protection for isolated workers.

Hotel management: With its first pilot deployments in the luxury hotel industry, Ofelia® integrates most of the concepts specific to the hotel industry, such as customer loyalty programs, hierarchical and configurable management of missions, quality-of-service analyses for in-room assignments, efficient search for qualified personnel closest to the intervention, etc.

Prisons: In this specific context, Ofelia® solution offers management of staff protection, snatch detection, voice connection, monitoring of rounds and temporal alarms, etc.

Events at the heart of the system

Notion of Requisition

The primary goal of the software is to proceed with the requisition of the most appropriate staff members for a given event. This requisition mechanism consists in calling by different means the most relevant personnel to respond to a given situation, according to scenarios defined in advance. The modularity of the system is based on the ability to react to multiple types of events, to define and unfold complex scenarios according to the environment, and finally to precisely identify the right people to contact according to the scenario but also according to the specific context of the event.



Notion of Event

The entire kinematics of the software is organized around a central notion of **event**. These events can have different natures (user assignments aka *missions*, nurse calls, technical alarms, system events, etc.) but are all manipulated in a totally unified way in the product. From the common dashboard to the centralized programming of scenarios and triggering rules, not to mention the consultation of statistics or the definition of alerts, the software user does not have to learn a specific way of working. Once familiar with the management of an event, his knowledge can be applied directly to all other types of events. Only some screens differ slightly in the presentation of specific information.

Natures of events

The event as such is made up of universal and immutable properties such as its creation date, its initial or current state, its participants, etc. It also has additional characteristics specific to its nature that allow access to its specific functionalities. The different **natures of events that** can be used in Ofelia® are the following:

- 
System events: These include all the technical events produced by the software itself, such as the loss of communication with a PLC, a fault on a driver, etc. These events can in turn be the subject of a staff requisition or start the automatic execution of alert scenarios.
- 
Technical Alarms: Alarms emitted by specific drivers: it can be dry contacts, detection of specific values on automats, PLCs or sensors, etc.

-  **Nurse calls:** These events take care of the specific management of call boxes in a hospital, they allow to manage calls, emergencies as well as the presence of the staff.
-  **Missions:** These events are elements that allow the management of the life cycle of a task assigned to one or many users. They can be maintenance missions in a hotel residence as well as maintenance follow-up assignments in an industrial environment. These missions can be triggered manually, at a given date and time or on a recurring basis.
-  **Runaways:** Runaways are events that are typically triggered by medallions in areas equipped with sensors. Their use is common in psycho-geriatric units (PGU) of senior residences.
-  **LWP (Lone Worker Protection) alarms:** LWP alarms are events whose life cycle depends on a mobile device (smartphone) associated with a staff member. The equipment detects situations of fall, immobility, loss of verticality, etc. to ensure the alert of the rescue teams and the geolocation of the victim.
-  **Notifications (or Mobilizations):** Notifications are fairly basic events that ensure the transmission of information with or without acknowledgement, with an extremely simple life cycle (Start / Success / Failure). They are mainly used as child events in mobilization campaigns or in gateways with other systems.
-  **Mobilization Campaigns:** The campaigns meet a need for massive and rapid notification, for example in the case of an emergency plan in a hospital. It is a way to group together a set of notifications for a single event and to monitor it globally. The campaigns are the subject of specific triggering and monitoring dashboards in the Ofelia® user interface.

Types of events

In addition to its nature, each event in the system has an **event type**. This type is a central notion in the entire logic of Ofelia®. Indeed, this type of event will determine the format of the messages that are sent to the requested users, the monitoring monitors associated with the event during its life, as well as various indicators and parameters. Each event, regardless of its nature, has a unique type around which all the rules governing the operation of the software are based. These types of events can be linked to user-defined custom properties that allow specific information to be collected and displayed, making it totally customizable according to the client's needs.

Hierarchy and inheritance

The notion of event type, like other Ofelia® notions (zones for example) is a hierarchical concept: all event types of each nature derive from an original type, and it is possible to specialize each type of event in an increasingly fine way by inheriting the characteristics of its parents over and over.

Thus, we can define a behavior on a type of event of Missions '*Maintenance*' that will apply on annual maintenance, revisions, and emergency interventions, until one of these elements modifies it for itself and its own children or adds custom properties.

 : values inherited from higher levels (Maintenance or Missions)

Properties

Each type of event can be associated with specific properties in the form of key / value (untyped).

Mot clé *	Description *	
Kilométrage	Nombre de kilomètres depuis la révision	☒ Obligatoire
+ Ajouter un paramètre		

These properties are inherited by successive additions to the child event types: they are accumulated on the children and grandchildren and cannot be removed from the hierarchy. When an event is created, these attributes can be filled in by the user and can then be used in the rest of the software (for example in the content of messages transmitted during requisitions).

Life cycle

Creation

The creation of a new event, depending on its nature and context, can be done in three ways:

- **Manual creation:** This is the case, for example, for missions created by a user or a regulator. The specific context of the event is then entered by the operator himself.
- **Scheduled creation:** Some types of events can be prepared by an operator and then their creation triggered at a later date and time, or according to a predefined recurrence pattern (*'Every first Monday of the month'* for example)
- **Creation by an event generator:** the creation is delegated to a technical process (sensor of an alarm button for example) or an algorithm according to a context or a variable of the system (for example if the temperature does not fall below 77°F for more than 10 minutes). The administrator has then programmatically defined the conditions that should trigger the generation of a new event in the system.

Finite state machine

Each event is represented in the form of a finite automaton, i.e., an abstract construction that can occupy a finite number of states and that is in a single state, called the *current state* of the event, at each moment. Each state change is materialized in the system by a *state transition* from state A to state B. Most of the logical rules for triggering scenarios and alerts, as well as the chronological representations of events, are based on these state transitions.

Depending on its nature, an event has a finite state machine with different transitional states. These different transition states are detailed here:

System Events

A system event has only two possible states during its life: Active and Inactive. It can switch many times between these two states until its final termination. Its initial state is always active.



Technical Alarms

Technical alarms allow you to manage several active levels, by assigning them a measure of importance or priority. Moreover, an alarm must first be acknowledged before it is reset (and therefore terminated).

Commenté [OB1]: stateDiagram-v2

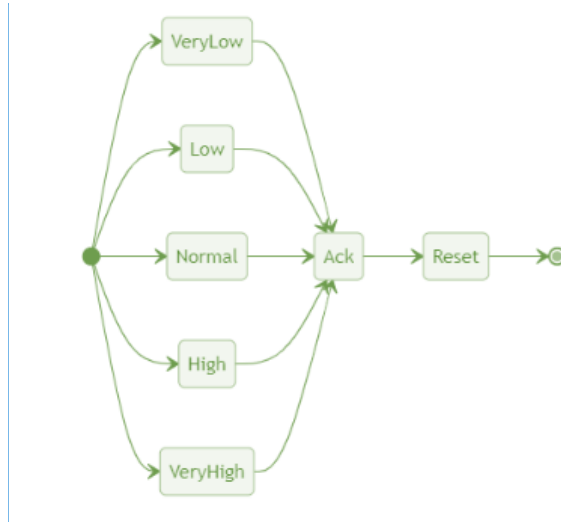
```

direction LR
[*] --> Active
Active --> Inactive
Inactive --> Active
Inactive --> [*]
  
```

Commenté [OB2]: stateDiagram-v2

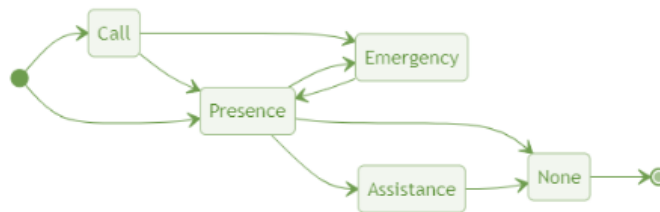
```

direction LR
[*] --> VeryLow
[*] --> Low
[*] --> Normal
[*] --> High
[*] --> VeryHigh
VeryLow --> Ack
Low --> Ack
Normal --> Ack
High --> Ack
VeryHigh --> Ack
Ack --> Reset
Reset --> [*]
  
```



Nurse calls

The life cycle of a nurse call is modeled on what is available with third-party systems such as ASCOM® or ACKERMAN®: in general, they begin either with the call made by a patient on the box or with the presence of a nursing staff in the room. Then, the call ends with the end of the presence of the last staff, after the event has possibly gone through the escalation phases of 'Emergency' or 'Assistance'. A nurse call also depends in its life cycle on the possible influence of other calls located in the same room and having a master-slave relationship with the main event.



Missions

The mission cycle is often dictated by manual status changes corresponding to the progress of a mission from its creation to its success or failure, possibly including pause and resume phases. In some cases, the administrator can configure a quality control validation phase for the type of Mission event before proceeding to its final closure, or define an automatic start upon its creation.

Commenté [OB3]: stateDiagram-v2

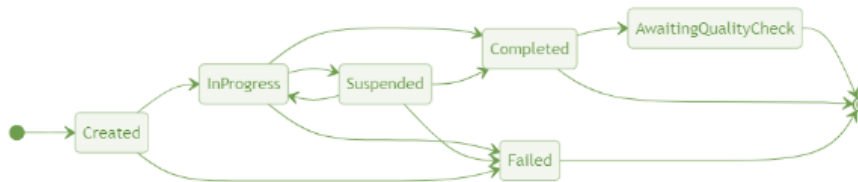
```

direction LR
    [*] --> Call
    Call --> Presence
    [*] --> Presence
    Call --> Emergency
    Emergency --> Presence
    Presence --> Assistance
    Presence --> Emergency
    Presence --> None
    Support --> None
    None --> [*]
  
```

Commenté [OB4]: stateDiagram-v2

```

direction LR
    [*] --> Created
    Created --> InProgress
    InProgress --> Suspended
    InProgress --> Completed
    InProgress --> Failed
    Suspended --> Failed
    Suspended --> InProgress
    Suspended --> Completed
    Created --> Failed
    Completed --> AwaitingQualityCheck
    AwaitingQualityCheck --> [*]
    Completed --> [*]
    Failed --> [*]
  
```



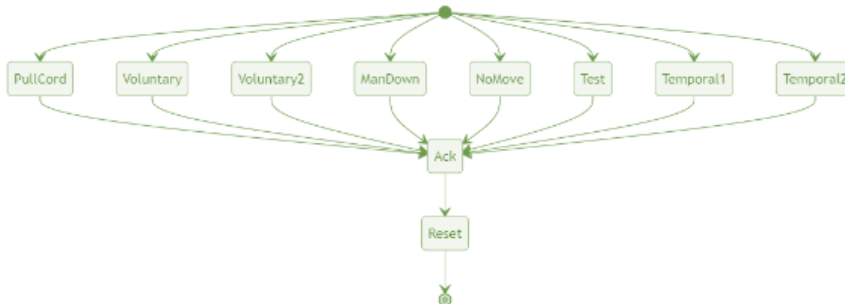
Runaways

The runaway has a particularly simple life cycle: once created, the only possible state is its termination. This absence of state change does not prevent a runaway in the 'Created' state from having its information evolving internally independently of its general state: thus, through the use of location tags, the history of the fugitive's instantaneous position and path are associated with the event without affecting its current state.



LWP alarms (Lone Worker Protection Alarm)

A LWP alarm has a state machine that is very similar to that of a technical alarm: instead of different alarm priorities, the LWP alarm has start states that depend on the nature of the trigger that caused the alarm to be created: cord pull, fall, automatic triggering on a time delay, etc. Beyond this differentiated start state, the rest of the cycle is identical to a technical alarm, with the need to acknowledge the event before it can be terminated.



Notifications/Mobilizations

A notification, once started, goes directly to a terminal state depending on the result of the notification: success, failure or cancellation are the second and last step of their finite state machine.

Commenté [OB5]: stateDiagram-v2

```

direction LR
[*] --> Created
Created --> Terminated
Terminated --> [*]
  
```

Commenté [OB6]: stateDiagram-v2

```

direction TB
[*] --> PullCord
[*] --> Voluntary
[*] --> Voluntary2
[*] --> ManDown
[*] --> NoMove
[*] --> Test
[*] --> Temporal1
[*] --> Temporal2
PullCord --> Ack
Voluntary --> Ack
Voluntary2 --> Ack
ManDown --> Ack
NoMove --> Ack
Test --> Ack
Temporal1 --> Ack
Temporal2 --> Ack
Ack --> Reset
Reset --> [*]
  
```

Commenté [OB7]: stateDiagram-v2

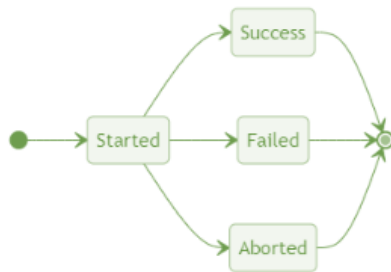
```

direction LR
[*] --> PullCord
[*] --> Voluntary
[*] --> Voluntary2
[*] --> ManDown
[*] --> NoMove
[*] --> Test
[*] --> Temporal1
[*] --> Temporal2
PullCord --> Ack
Voluntary --> Ack
Voluntary2 --> Ack
ManDown --> Ack
NoMove --> Ack
Test --> Ack
Temporal1 --> Ack
Temporal2 --> Ack
Ack --> Reset
Reset --> [*]
  
```

Commenté [OB8]: stateDiagram-v2

```

direction LR
[*] --> Started
Started --> Success
Started --> Failed
Started --> Aborted
Success --> [*]
Failed --> [*]
Aborted --> [*]
  
```



Mobilization Campaigns

Since a mobilization campaign is only intended to serve as a host for a set of child notification events, its lifecycle consists of only two states: Started and Completed. It is at the level of each notification that the notions of individual success or failure are held, the campaign itself having no such status.



Commenté [OB9]: stateDiagram-v2

direction LR

[*] --> Started

Started --> Terminated

Terminated --> [*]

Global state priority

In order to allow a unified representation of the various events according to different projections (by service, on a location map, etc.) a notion of *priority of importance* is defined in a global way regardless of the nature of each event.

Depending on the context of the installation, each administrator can define what he/she considers to be the most relevant to prioritize in the displays. For example, if several events of different natures and types occur in the same geographical area, the administrator can choose to assign the color of the type and status that seems to have the highest priority to the area: it is often more relevant to display a color indicating that a floor is affected by a fire than to display the fact that a mission has just ended in this location.

OFELIA DÉFINITIONS DES ÉTATS		
Différence des états		
Icone de l'événement	État	État
Alarme DATA	Volontaire	
Système	Actif	
Alarme DATA	Volontaire 2	
Alarme DATA	Anchorage	
Alarme DATA	Perte de verticalité	
Alarme DATA	Immobilité	
Alarme	Alarme (très haute)	
Alarme	Alarme (haute)	
Alarme	Alarme (normale)	
Alarme	Alarme (basse)	
Alarme	Alarme (très basse)	
Fugue	Crise	

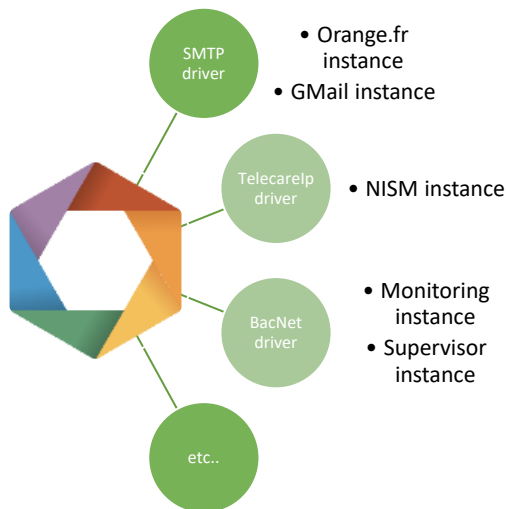
This prioritization of states is all the more useful to take into account the hierarchical nature of geographical zones when the user displays, for example, the ground plan of a building, then navigates from floor to floor until the level of individual rooms: it is the most relevant child event that must in this case be made visible to the operator regardless of the selected zoom level.

Connections to the information system

Drivers and driver instances

Communication between Ofelia® and the external environment is carried out using specific drivers supplied by Ascom. These drivers connect the software with various communication protocols, in order to ensure *inputs* into the system (sensors, exchange protocols, etc.) and *outputs* (sending calls, SMS, changing PLC values, etc.).

More precisely, the **drivers** encapsulate all the logic and implementation of the communication protocols, and the Ofelia® administrator deploys **driver instances** that use these drivers with a specific configuration (network address, user, password, etc.). The same driver can be deployed in the form of several driver instances on the same Ofelia® deployment: this will be the case, for example, if the installation is connected to several PLCs using the Modbus® protocol, or several e-mail servers.



Capabilities

The list of drivers available on a given installation depends on the license granted to the user site. Each driver is able to fulfill one or more functionalities ("Capabilities") which govern their use in the product: call emission, variable manipulation, etc. These capabilities are described below:

Capability	Functionality
Caller	Making outbound calls to devices
Alarm	Receiving alarms
Interactivity	Dialogue via interactive screens (menus, questions and answers, etc.)
NurseCallRead	Reading the status of a nurse call
Availability	Provision of an availability report
Variable	Manipulation of variables (programming, updating, browsing)
Action	Task list management
SystemEvent	Generation of system events (in case of failure for example)
Escape	Follow-up of runaways
LwpAlarm	Protection of the isolated worker
NurseCallWrite	Writing the status of a nurse call
UserLocation	Geolocation of users
NurseCallVirtual	Ability to manage nurse calls without a dedicated box
Template	Mobile Device Configuration Template
Text	Receiving frames in text format

Drivers provided with Ofelia®

The following list shows the main drivers available in the application, allowing you to configure one or more parameterized instances for each one:

Driver	Description	Features
 <i>Ascii</i>	Receiving text frames on a TCP socket	• SystemEvent • Text
 <i>Ascomlp</i>	Dialogue with an Ascom IP-based central	• SystemEvent • Text • Caller • LwpAlarm • Interactivity • Availability • NurseCallWrite • UserLocation

 <i>Ascomlp OneWay</i>	Sending simple messages (without acknowledgement)	• Variable • Caller
 <i>BACnet</i>	Reading variable values on a BACnet server	• Variable
 <i>DigitalVirgo</i>	Sending SMS	• Caller
 <i>Email</i>	Sending emails via SMTP	• Caller
 <i>Espa</i>	Receiving frames in ESPA format	• Text
 <i>HighSMS</i>	Sending SMS	• Caller
 <i>MiniFinder</i>	Lone worker protection with miniFinder beacons	• LwpAlarm • Availability
 <i>ModBus</i>	Reading variable values on a Modbus server	• Variable
 <i>MultiTech</i>	Sending SMS on GSM modem	• Caller
 <i>NurseCall Virtual</i>	Virtual nurse call management	• SystemEvent • NurseCallRead • NurseCallWrite •
 <i>Ofelia Messenger</i>	Management of the windows messaging application	• Caller • Availability
 <i>OfeliaSmart</i>	Dialogue with mobile devices equipped with the OfeliaSmart application	• Caller • LwpAlarm • Availability • UserLocation • Template • Variable
 <i>OPC UA</i>	Reading variable values on an OPC UA server	• Variable
 <i>Opera</i>	Synchronizing clients from an Opera server	• Action
 <i>Ping</i>		• Variable
 <i>PopUp</i>	Dialogue with mobile devices equipped with the Mercury Pop-up application	• Caller

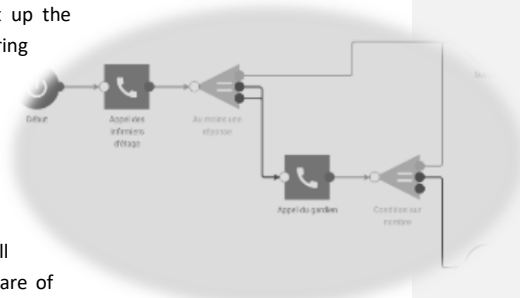
 SIA	Sending messages to the AIS alarm system	• Caller
 SIP	Sending voice messages to a SIP exchange	• Caller • Text
 SMSFactor	Sending SMS	• Caller
 SMTP Server	Receiving text frames via SMTP	• Text
 Stanley	Stanley proprietary lone worker protection system	• LwpAlarm
 ascom Telecare IP	Management of nursing calls and runaways in Telecare IP format	• SystemEvent • NurseCallRead • Escape • Variable
 ascom Telecare M	Management of nursing calls and runaways in Telecare M format	• SystemEvent • NurseCallRead • Escape •

Triggers and Scenarios

Workflow engine

The definition and execution of requisition scenarios in Ofelia® is ensured by a workflow engine. It allows an administrator to easily set up the execution flow he wants to perform when specific triggering conditions are met.

The configuration is done in two steps: on the one hand the creation of the different execution workflows, and on the other hand the definition of the triggering rules leading to their execution. The creation of a workflow is therefore as independent as possible from the event that will lead to its execution: it is the triggering rules that take care of analyzing the particular context of the triggering to find the most relevant workflow to execute.

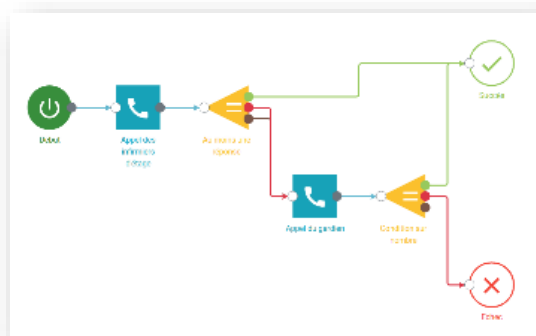


Definition of workflows

The definition of execution scenarios is based on the logical sequence of activities representing actions to be performed (e.g., making phone calls or updating system variables) and logical tests (i.e., are there still users in the area? Have enough people been requisitioned? etc.).

To elaborate his workflow definition, the administrator has a palette of actions and conditions allowing to craft complex cascading sequences according to the evolution of the requisition: for example, calling a second group of users if the first one did not allow to requisition enough people, triggering a siren, sending an SMS to an administrator, etc.

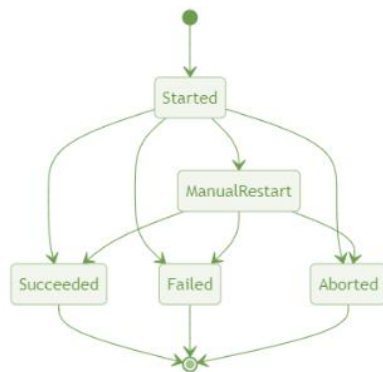
The following example shows a simple call cycle whose purpose is to call a floor nurse and escalate the call to the guard if no one has accepted the call after a certain amount of time, of course it's possible to create much more complex scenarios:



State machine

When a workflow is initiated, it executes and suspends when conditions do not allow it to continue. When the context changes (e.g., a user has responded positively to a request, a deadline to respond has expired, etc.), the workflow is re-evaluated and resumes until it is suspended again or reaches one of its endpoints.

A **workflow definition** is the plan defined by the administrator describing the steps to be performed and their sequence. When a trigger rule meets the necessary conditions, it creates a **workflow instance** which is a functional copy of this definition, associated with the execution context of the event. This copy has a global progress state as well as a current state for each of its nodes. A workflow instance must end with one of the three states "Success", "Failure", or "Cancelled".



Commenté [OB10]: stateDiagram-v2

```

[*] --> Started
Started --> Succeeded
Started --> Failed
Started --> Aborted
Started --> ManualRestart

ManualRestart --> Succeeded
ManualRestart --> Failed
ManualRestart --> Aborted

Succeeded --> [*]
Failed --> [*]
Aborted --> [*]
  
```

Workflow activities

The nodes of a workflow that trigger actions are called **activities**. The activities "Start" and "End" (either success or failure) are mandatory activities in each workflow. For a better readability, each activity can be named freely.

Start



This mandatory activity is placed automatically by the workflow editor. It can be renamed but not deleted.

Only one Start activity is possible, it is the **only** starting point of the execution.

Call



This activity allows to make calls to users' media. It includes specific parameters (chosen strategy and call profile, etc.) as well as a set of requisition rules. These rules (see on page 31), allow to define a set of users to include or exclude from the calls.

Each call activity can be associated with a 'Call Group' to allow writing complex conditions taking into account the execution paths.

Calling while a zone is occupied



This activity is identical in design to the call activity but has an additional parameter: it continues the call for a maximum duration or as long as there are people present in the event area. Of course, this activity can only work properly if user location devices are set up at the site and in the software.

Waiting for all calls



This activity forces the requisition engine to wait until all previous calls have been transmitted before continuing execution. A maximum wait time can be defined, resulting in a conditional branch.

Wait for a specific time



This activity forces the requisition Engine to wait for a certain amount of time before resuming execution. Once the wait time is over, the workflow execution proceeds normally.

Empty the list of users



This activity cancels the requisition of staff who have already been requisitioned in the workflow. It is useful, for example, if, due to an insufficient partial condition, you want to requisition a completely different type of staff and not the few who had already agreed.

Variable writing



This activity forces the writing of a value to one of the system variables. It can be a way of communicating with an output device (powering a siren for example) or of causing other workflows to trigger in cascade (if an event generator is listening to the changes this variable)

Delete messages



This activity cancels all calls still in progress for the current event. It is useful for example if a new workflow is configured to start on a failure condition and you want all the calls made to be deleted before proceeding with a new requisition.

Success, Failure



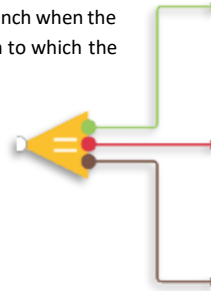
These activities are the end nodes of the workflows. When the execution engine reaches these nodes, the workflow enters the '*Success*' or '*Failure*' state and stops. A workflow can have several Success and/or Failure nodes, each connected for readability purposes to different branches.

An optional parameter of these end nodes allows you to force the event state to change to a given value when they are reached.

Workflow conditions

The nodes of a workflow that allow logical branches to be made and thus condition the continuation of the execution of the scenario on a contextual value are called **conditions** of a workflow. The branching conditions mainly concern a way to count the number of users who have been requisitioned (i.e., who have accepted a call).

Condition nodes have three output branches to other nodes in the workflow: a branch when the condition is met, a branch when the condition is not met, and an optional branch to which the execution continues if a predefined timeout is reached. This allows to choose another path if the expected number of people has not been reached after x seconds for example.



The following conditions are available:

Condition on number



This condition ensures that exactly the right number of people has been requisitioned. As soon as the number is reached or, on the contrary, if the number of users able to answer cannot reach the configured value, the execution continues in success or failure. If a greater number of people responds favorably almost simultaneously (before the calls have had time to be cancelled), only the first users to respond will be requisitioned.

Condition on minimum number



This condition ensures that a minimum number of people has been requisitioned. As soon as the number is reached or, on the contrary, if the number of users able to respond cannot reach the configured value, the execution continues in success or failure. If a greater number of people responds favorably, all these users will be considered to have been requested.

Condition based on a script



This condition allows to define a logical expression (and, or, =, >, <, etc.) using the concept of a *Call Group*. It allows to define complex conditions based on the number of people requisitioned in different branches of the workflow (by counting the number of people per call group and not only on the whole workflow).

Requisition rules

Each call node lets the user define a set of requisition rules to include or exclude one or more users based on different criteria. When the workflow instance is executed, the set of rules in this list is evaluated according to the current context (date and time, location of the event, known location of staff members,

etc.) to build a list of people to call. Requisition rules can be inclusive or exclusive so as to allow for scenarios such as *"Call all maids in the area except Maria"*. Exclusions are always evaluated *after* inclusion rules.

Requisition Mode

The rules have several selection modes, associated with contextual parameters, as well as additional contextual filters. The modes are as follows:

- **Call a staff member:** Requisition of a specific user.
- **Call the employees assigned to a function:** Calls all the users assigned to the requested function at the time the activity is executed. This mode takes into account the staff schedules and possible absences or on-call duty.
- **Call the employees assigned to the default function of the event:** Each type of event can be associated with a function (i.e., a job) to be requisitioned. In this case, the users occupying this function at the time of the execution are called. This mode also allows to define a fallback function (if no default function is associated with the current type of event) and a behavior in the event of the absence of these two functions (default and fallback): either call nobody or call all the employees present.
- **Call the creator of the event:** This mode can be useful for example to send an acknowledgement message or a notification of acknowledgement in the case of a LWP alarm triggered by a staff member.
- **Call the users of the departments of the event creator:** allows to requisition on-site employees who belong to the same department as the creator, for example the members of the same department as the user who has triggered an alarm for the protection of lone workers.
- **Call the initiator of the state change:** This mode is similar to calling the creator of the event, but it extends to the person who initiated any state transition of an event (not just its creation). This could be, for example, the person who acknowledged an alarm.
- **Call requisitioned employees:** This mode is particularly useful when sending a confirmation or instruction in a notification workflow, once the requisition has been completed. It targets only those staff members who have accepted the requisition in a previous workflow.
- **Call the staff assigned to the area:** This involves contacting the staff members in charge of the area where the event occurred. Depending on the additional filters (see below) it can be the exact area of the event or the floor or another level even more encompassing.
- **Call the employees present in the area:** This call exploits the presence and geolocation data of the users according to the capabilities of the drivers deployed on the site. As before, depending on the additional filters, it can be the exact zone of the event or a higher level in the hierarchy (people in the room or on the floor or in the building for example).
- **Calling non-requisitioned employees:** This is the reverse of the "Calling Requisitioned Employees" mode: this time the call is made to staff members who were previously contacted for the event and who refused the call or accepted it too late.
- **Calling the requisitioned and non-requested employees:** this mode allows you to call all the people already contacted for the event, whether they have accepted the call or not.

Requisition filter

Once the mode is defined, additional filters can be added that modify the default operation of the requisition mode. These include further restricting the list of people to call or changing the selection context.

List Restriction

- Assigned to the service area of the event
- Assigned to a department of the event creator
- Assigned to the area
- Present in the area

Geographical selection context: When using a geographical mode or filter (for example: "call the staff present in the event zone"), it is possible to exploit the fact that the geographical zones of the facility are defined in a hierarchical way (for example *Building A / Floor 1 / East corridor / Room 214*) by indicating to go up a certain number of levels and to include or not include the child zones in the selection. Thus, if the staff is assigned to a 'Corridor' level for example, you can choose to go up one level when an event occurs in a room. In the case of a room fire alarm, you can similarly indicate to go up 2 or 3 levels and to include the children's zones if you want to call all the staff on the floor or in the building.

Triggering rules engine

Once the execution scenarios are defined in each workflow definition, it is necessary to indicate the conditions that will trigger the execution of each of these scenarios. The tool used to define these conditions is Ofelia's *Triggering Rules Engine*. For a given type of event and a given type of state transition, it defines a set of conditions to be met in order to trigger a particular workflow.

Trigger types

A trigger rule is specific to a type of event (Mission, Alarm, etc.) and to a change evaluation mode. Indeed, the launch of a workflow is conditioned by the transition of an event (between two states, two priorities, etc.), and different triggering rules can be created depending on the type of transition that is evaluated:

- **On a change of state of the event:** These rules are evaluated each time an event transitions its current state. For example, when an alarm goes from the *acknowledged* state to the *reset* state. For a list of transitions of each kind, see on page 18.
- **On a change of state of the request:** These rules are evaluated each time the state machine of a workflow instance changes. For instance, when a workflow changes from *started* to *cancelled*. For a list of workflow states, see on page 27.

- **On a change of priority:** These rules are evaluated when the priority of an event is changed by the user, they allow for example to select a new workflow when an event goes into a more urgent state. For a detailed explanation of how priorities work, see *Event Priorities* on page 49.
- **On a change of zone:** These rules are essentially linked to the *Runaway* nature of events. They allow to trigger a different workflow when an event changes its current zone. In the case of a runaway, it is possible to trigger a different scenario depending on the zone in which the fugitive arrives (simple notification as long as he remains in the department, for example, and a call to security if he leaves further).

Trigger conditions

Once the nature of the event and the type of transition have been defined, the administrator can fine-tune the precise conditions for triggering a rule: this is a set of properties that must *all* be met for the rule to be considered active.

Order of precedence

The triggering rules for each type of transition are sorted by an order of priority defined by the administrator: when an event occurs for this type of transition (status, priority, etc.) the rules are then evaluated sequentially according to this order. As soon as a rule has all its conditions met, the corresponding workflow is executed and the following rules are not evaluated: for a given event transition, only one triggering rule can be triggered and consequently only one workflow can be active at a given time on this state transition.

Specific criteria

At a minimum, a rule has a name, a precedence order, and a workflow to trigger. In its strictest expression, the specified workflow is executed whenever an event of the target nature makes the expected type of transition (change of state, requisition, priority, or zone).

In order to precisely refine the workflows to be started according to the context, advanced filters are available for each rule. These filters, aimed at precisely restricting the scope of the rule, are as follows:

- **Event types:** List of event types on which to restrict the application of the rule. The notion of event type hierarchy (see *Hierarchy and inheritance* on page 16) obviously applies to this criterion: Thus, a rule that applies to one type of event will also apply to all its child event types. It is therefore important to ensure that the rules are ordered in such a way as to place the most specialized rules at the top of the list, while possibly keeping a generic rule for the root event type at the end of the list.
- **Zones:** List of zones on which to restrict the application of the rule. As with the event types, the notion of hierarchical zones applies (see *Inheritance* on page 45): a rule for a floor, for example, will apply to all the rooms on this floor. Here again, the administrator should make sure to start with the most restrictive rules (at the level of individual rooms for example) and end with the most global ones.
- **Zone types:** List of zone types (e.g., room, hallway, floor, etc.) to which the rule applies exclusively.
- **Tags:** The rule applies only to events located in a zone whose type includes one of the specified tags. In the future, when the concept will be extended to other elements than the zone types, the rule will be able to filter on all these tags.

- **Departments:** List of departments (or *Medical Units* in healthcare) on which the rule applies: only the events located in a zone of one of these departments will be able to trigger the rule.
- **Priorities:** List of priorities of events concerned by the rule. This filter allows for example to trigger a specific workflow only when an event is *urgent*.
- **Beacons:** list of detection beacons associated with the event. Thus, in the case of a runaway for example, a scenario can be started when the fugitive passes a particular sensor.
- **Default functions:** Each type of event can be associated with a default function/job (possibly inherited from its parent). This filter allows the rule to be triggered only if the event processed has a type associated with one of the specified functions, for example only on missions processed by personnel with the function '*Maid*'.
- **Number of executions:** A workflow can be triggered consecutively several times for the same event, for example following the activation of a reminder monitor (see *Reminder Monitors* on page 47). This filter allows the rule to play a different scenario after a certain number of executions.
- **Roles within the event:** List of User/Role combinations needed to trigger the rule. This is to restrict the rule to events where named users occupy a specific role. Identifiable roles are *Creator*, *Requester* or *Recipient*: Thus, a rule may only apply if a mission is created by Mr. X and destined for Ms. Y.

Workflow parameters reported in the rule

Some workflow activities have parameters that are set by the administrator. These parameters can be, for example, the state that the event should take in case of success or failure of the workflow, the waiting time before considering a condition as failed, etc. These parameters can be set directly with a value in the workflow definition, or replaced by a variable whose value will be indicated in the triggering rule. In this way, the same workflow can be shared in different scenarios, simply by deferring the information of its execution parameters in the rules that trigger it according to the context.

User management

Ofelia's user management is based on a centralized approach that includes both *staff members* and, depending on the functional context and the license granted, *patients* and *customers*. Around a common base (mainly civil status like gender and name), each of these concepts has its own specificities.

Patients

Patients are users who may be associated with a protective alarm device, in the form of a runaway medallion for example. Ofelia® is not designed to manage the concept of patient to any great extent, so concepts such as Social Insurance Number, medical information or care records will not be included in the standard offer. For this type of scenario, the solution will interface with on-the-shelf solutions via gateways or external applications (see *Third-Party Applications* on page 86): these will create patients in the system with minimal information (Last Name, First Name and preferred language) and then create links to a referent patient database. Depending on the system in place, the information stored in Ofelia® can also be pseudonymized or even completely anonymized (for GDPR compliance).

Customers

The concept of *Customer* as used in Ofelia® is mostly targeting the field of hotel facilities. As with patients, only the minimum civil status information (Last Name, First Name and Preferred Language) can be edited in the software. In most cases, this information comes from a gateway with an external hotel management system (see *Opera®* on page 233). In addition, a simplified management of loyalty programs is included in the product, allowing a customer to be associated with such a program and making this information accessible in the message formats (in order, for example, to inform the staff of the 'VIP' status of a customer before entrusting him with an assignment).

Employees / Staff members

This is the most important type of user in Ofelia®. It represents a human resource that can be assigned one or more media (telephone, email address, etc.) and that can be requisitioned. Moreover, it is the only type of user that can have an account authorizing it to connect to the software via the web interface or a dedicated mobile application.

An employee may be associated with a lone worker protection system (LWP) and assigned to areas or departments of the facility. Finally, they may also have specific functions (jobs) on their own schedules (for example, acting as a guard during the week, and as a security referent during on-call periods every other week).

Department / Units

A department (similar to a *Medical Unit* in hospitals) is a grouping of areas. An employee can optionally be attached to one or more departments. This attachment allows you to influence the way requisitions are made (for example, calling all the caregivers in the same department as a nurse call event). This link can also influence the dynamic appliance of the user's restrictions (for instance the permission to edit only the zones of the units the user belongs to).

Presence Type

For the specific management of nurse calls (in particular via a *Telecare*® driver), the staff present in a room can identify themselves either as a nurse or as a doctor. The administrator can create as many presence types as he wants and indicate for each employee the type that corresponds to him. Thus, when this employee is present during a nurse call, the appropriate presence type and the right title will be indicated in the various visualizations of the event.

Functions

The function of an employee corresponds to his job (or role/occupation) at a given time. Usually, an employee has only one job (for example 'Doctor', 'Nurse', 'Guard', etc.) but it is possible to configure several jobs for the same user, either concurrently or each applicable to predefined schedules.

Planning

In order to fit the administrative organization of the facility as closely as possible, the function of an employee can be applied in a global way (full time assignment, whatever the day and time), or according to a calendar and hourly schedule.

Assigning an employee to duties in predefined time slots is done in two steps:

1. Creation of the schedule
2. Association of the function on each time slot of the schedule

Creation of a schedule

A schedule corresponds to a recurrence of days that are repeated according to a specific periodicity. The types of periodicities available are:

- **Daily:** A typical working day is defined (for example with two time slots *9am->11am* and *2pm->6pm*), and it is reproduced every day of the year in an identical way.
- **Weekly:** A typical working week is defined (e.g., *9am->11am* and *2pm->6pm* slots from *Monday* to *Friday*, no slots on weekends), and this is reproduced identically every week of the year.
- **Fortnight:** The administrator defines the functioning of a fortnight of work, and this one is reproduced identically every two weeks during the year.
- **Monthly:** The hours of each day of the month are defined in a precise way, this organization of work is then applied to each month of the year.

On these planning templates, the administrator starts by defining the types of time slot that will apply (for example: *Morning* and *Afternoon*), then he defines the time slots by coloring them on the planning with the desired type. To each time slot (start time -> end time) corresponds a type of slot and an optional information indicating that this slot is an on-call or an absence.

Nom

Types de créneau horaire : ☒ Matin ☐ Après-midi

Heure de début: 8 Heure de fin: 18

	lundi +	mardi +	mercredi +	jeudi +	vendredi +	samedi
08:00	☐	☐	☐	☐	☐	
09:00	☐	☐	☐	☐	☐	
10:00	☐	☐	☐	☐	☐	
11:00	☐	☐	☐	☐	☐	
12:00						
13:00						
14:00	☐	☐	☐	☐	☐	
15:00	☐	☐	☐	☐	☐	
16:00	☐	☐	☐	☐	☐	
17:00	☐	☐	☐	☐	☐	
18:00						

Absences and on-call duty

When a user is assigned to several schedules, this precision will be used to remove inconsistencies: on-call duty takes priority over absences, which themselves take priority over standard assignments.

Example

John works as a guard on a 'standard work week' schedule from Monday to Friday from 7:00 pm to 11:00 pm. If an alarm requiring the presence of a member of the guard group occurs on a Wednesday at 9pm, he will be called. However, if the alarm occurs on a morning or on a weekend, the system will not try to reach him.

Subsequently, John is assigned a second fortnightly schedule indicating that he is absent on the second Wednesday of each fortnight (i.e., every other Wednesday). When the alarm is triggered on the 2nd Wednesday of the month at 8pm, John is not called by the requisition engine: his absence has priority over his usual assignment.

Finally, John is also associated with the on-call schedule indicating that his presence is required on Wednesday 12 January (2nd of the month) all day, as well as on other specific days each month. If an alarm occurs on this particular Wednesday evening during his on-call period, he will be requisitioned despite his usual rest on this time slot: his on-call schedule has an even higher priority than the absence schedule.

Association of the employee with the schedule

When schedules are defined, each employee can be assigned to a given function on a type of slot of a schedule. For example, an employee can be directly assigned to the function 'Guard' on the slots of type 'Morning' of the schedule 'Standard week'.

Assignments, outside of any schedule, can also be added on an ad hoc basis to manage assignments specific to a single employee (e.g., absence for time off or leave, exceptional on-call duty, etc.).

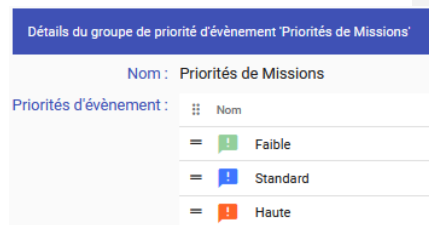


Availability

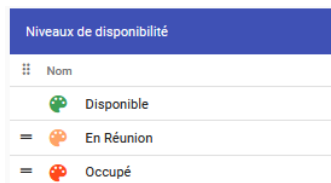
The management of the schedules allows to address simply the problem of the presence and on-call hours of the various employees, by restricting the scope of the requisitioned persons according to their schedule. However, this management is not sufficient to manage the real availability of the personnel present. Indeed, an employee can be present, his main media connected, but he can also wish not to be disturbed during a given period. More precisely, he may wish to be disturbed *only for urgent events* or events considered as such.

In order to allow for this relative level of availability depending on the nature of the events (even in a meeting, you may want to be called in case of a fire or a gas leak, but not for a simple end-of-mission notification), three concepts are present in Ofelia® :

- **Event priority:** The administrator creates priority groups each with several priority levels (for example a 'Mission Priority' group with 'Low', 'Standard', and 'High' levels). Each type of event can be associated to a priority group or inherited from its parent: this group defines the various priority levels that the user can associate to an event.



- **The user's immediate availability status:** the administrator defines different possible immediate availability statuses for users. By default, the 'Busy' and 'In a meeting' levels are provided. The 'Available' level is also integrated in the product and is subject to a specific treatment.



- **The availability profile applicable to the user:** By combining the two previous elements, the administrator defines *Availability Profiles* indicating for each priority group how calls should be made according to the level of availability of each user. For example, a profile can be defined that states that a 'Busy' user will not receive any calls for an event whose priority group is 'Mission Priority' while a user whose status is 'In a Meeting' will only receive calls for the same type of event whose priority is at least equal to 'High'.

Nom : Profil par défaut

Disponibilité de l'utilisateur	Groupe de priorité d'évènement	Appeler l'utilisateur
En Réunion	Priorités de Missions	Si l'évènement est au minimum  Haute
Occupé	Priorités de Missions	Jamais

The administrator then associates the desired availability profile to each employee in order to manage the expected behavior according to the availability status of each employee.

Designations

Nom : Stan Laid

Langue préférée :

Activer la connexion permanente : Non

Type de présence :

Profil utilisateur : Profil par défaut

Profil de disponibilité : Profil par défaut

Capacité DATI : Oui

Alarme DATI : Suivre capacité du média

Type d'alarme DATI :

Thereafter, the user is autonomous to modify his level of immediate availability, either through the web interface or via the mobile application. The call engine will then be able to act appropriately according to the priority of each event occurring in the system.

Général Paramètres

Profil de disponibilité : Profil par défaut

Niveau de disponibilité *

Disponible

Disponible

En Réunion

Occupé

✓ Valider

Login ID

One of the specificities of the employee (contrary to the other categories of users such as Customers or Patients), is the possibility to authorize him to connect to the product. This connection can be done from one of the compatible clients (currently the Ofelia® web interface or the OfeliaSmart® mobile application).

To achieve this, the administrator associates the employee with a user account in one of the security providers available in the product. By default, when Ofelia® is installed, it deploys its own security system (username/password) but other providers can be added in the form of external applications (see *Extensibility and interfacing* on page 78), for example the LDAP/Active Directory provider (see *LDAP Directory Application* on page 247).

Communications and calls

The communication between Ofelia® and the users goes through devices interfaced with different drivers. As a standard feature, the software allows a number of media (DECT phones, pagers, etc.) to be assigned to each employee in order to send them messages. In addition, specific devices are provided to feed the system with real-time information, this time as input: this is the case for example with nurse call boxes, patient medallions, or even location beacons.

Media

Each media is identified by its model and address. It can be self-service (and assigned to a user on demand, when the user picks it up from a rack) or assigned to a specific user.

Media Types

Each media type is identified in the system by a specific template. This feature makes it easy to identify and assign a specific behavior to a set of media by specifying, for example, to which type of user it can be assigned (employee, patient or customer) and by allowing the configuration of all media of the same type using a single template (see on page 42).

A media type allows to group the configurations of a set of media by defining for each driver instance the associated message types.

Addresses

Each media defines a unique address for each configuration of its model (i.e., for each *driver instance/message type* pair). When the media is called on this address, Ofelia® takes care of formatting and sending the appropriate message format to the media.

Call profile

A call profile is a named configuration that allows to define a specific behavior for each driver of the system. Thus, each driver exposes a certain number of its own parameters (e.g., the number of beeps, the color of the text, the acknowledgement delay, the possibility or not to handle acknowledgements, etc.) and the profile allows to configure once and for all the behavior for all driver instances. When such a profile is selected (by means of a strategy for example), calls to all media, whatever their driver, will be made with a consistent behavior.

For example, we can imagine an 'Emergency' call profile defining for each driver a behavior with an audible warning, resumption during 5 attempts, etc.

Détails du profil d'appel 'Urgence'

Nom : Urgence

Email

Le profil d'appel utilise la configuration suivante :

Priorité : Urgent Importance : Haute

IMS3

Le profil d'appel utilise la configuration du profil d'appel par défaut pour cette instance de driver

Type d'alerte : 2 bips Couleur : Rouge Priorité : Priorité haute (4)

Accepter / Rejeter : Accepter et rejeter Attente acceptation (sec) : 300

☐ Rejet automatique si pas d'acceptation

☒ urgent

Valider Annuler

Call Strategy

To define the global behavior of a call made to a user, the software has a notion of strategy. Strategies are named elements in the system that define the behavior of a call to a device. These strategies can be specified in the *Call* activities of the workflows and define the call profile to be used, as well as the number of attempts to be made and the time to wait between two attempts.

Media templates

Media templates are a configuration option reserved for a certain type of drivers supporting the customization capability. They are closely related to drivers that support mobile applications: they force device configuration and authorizations on phones from the Ofelia® application without having to manually configure each device.

Détails du template de média 'Route'

Nom : Route Configuration de périphérique :

Le template de média utilise la configuration par défaut pour cette instance de driver

Configuration du serveur

Paramètres généraux

Configuration des alarmes

Volontaire

Volontaire 2

Perte de verticalité

Immobilité

Perte de connexion

Test

Configuration des localisations

BLE

WiFi

Extérieure

Configuration du serveur

Port de connexion au driver du serveur : 30240

☐ Activer HTTPS

Port de connexion à l'API : 80

Mot de passe d'administration :

Editer

Message format

In order to send the most relevant message to each user, Ofelia's[®] requisition engine is based on pre-formatted messages defined by the administrator. These messages can be defined for each type of event. Given the hierarchical nature of event types, in its simplest configuration, the administrator determines a message format for each state of each nature of event, and this format will be applied to all events of this type. If he later wishes to specialize a message format for one of the states of a specific event type, he will only have to modify this message format: the others will remain unchanged, inherited from the default one.

Format	Condition
@EventType @EventState<CR> @Zone<CR> @RequestedFor<CR>	
@EventType @EventState @RequestedFor<CR> @Zone<CR>	MessageTypeId == Défaut AND EventState == Actif
@EventType @EventState @RequestedFor<CR> @Zone<CR>	MessageTypeId == Défaut AND EventState == Inactif
Nouvelle priorité @Priority pour l'événement @EventType en @Zone	MessageTypeId == Défaut AND StateTransitionKind == Sur changement de priorité

Each message format is associated with a logical condition indicating when this format must be applied. The different formats inherited for an event type are then evaluated in order until a condition is verified. It is then this format that will be applied and sent to the user, once the different contextual variables have been replaced. Each message can therefore contain dynamic information generated at the very moment the message is sent (for example, how long ago the event was generated, its priority level, the zone where it is active, etc.).

Devices

While the media are mainly used to send messages outside the system, other devices are used to provide information *into* the system. Some of them are provided by default depending on the license of Ofelia® deployed on the site: for example, the nurse call boxes or the runaway medallions.

Nurse call boxes

The administrator must declare each nurse call device (or import the list) and then indicate which driver instance controls it and in which room of the facility it is located. This management, coupled with the support for hierarchical zones, will allow the handling of a relevant requisition of the nursing staff during calls (by department, by floor, etc.).



It is also possible to indicate for each box what type of event (of a 'nurse call' nature) it generates: it is thus possible to imagine precise scenarios triggering, for example, a dedicated workflow for emergency calls in a resuscitation room, or a specific treatment for a given department of the hospital.

Medallions



Just like for the nurse call boxes, a special management is planned for the handling of the patient medallions. In this case, each medallion is associated not with a room, but rather with a patient. Here again, it is possible to define for each medallion which driver instance controls it and which type of escape or nurse call event is generated in case of a runaway or a call.

Beacons

Beacons are also part of the devices that feed the system with real time data. They are managed in a special way, allowing the administrator to associate a beacon with a zone (i.e., not only a room, but also a hall, a corridor, etc.). Each tag is associated with its own internal identifier and the administrator can adjust its level of precision. He can also define a beacon as '*excluding*', which means that the detection of this beacon causes the temporary stop of the localization process: this option is particularly useful near the exit doors of a building to indicate to the system that a localization by beacon is no longer necessary/relevant as soon as a user is detected outside the premises (in this case, we will switch to another localization mode, by GPS for example).

Geographic locations

A central element of the software's configuration is the representation and structural division of the client facility into buildings, floors, rooms, corridors, etc. This architecture allows to

- Associate staff and detection devices (sensors, nurse call boxes, etc.) with zones
- Define the behavior of the requisition engine according to the zones and their position in the tree
- Display events in real time on plans, with the possibility to zoom from level to level, from the ground plan down to the floor plan for example

The objective of this organization is to have a geographical division in the form of a tree structure, in order to allow a parameterization from the broadest to the finest.

Zones, zone types and positions

In the internal representation of the facility, each individual area is considered a "zone". These zones can be of different types, which can be used to determine other aspects of the software's behavior: for example, triggering certain scenarios only if an event occurs in a room, or requisitioning all the personnel assigned to the same floor as an alarm, etc.

One concept allows each type of zone to be qualified even more precisely: assignable positions *inside* the zones. A distinction is made to identify the zones in which it is possible to assign people (patients, residents, etc.) or peripherals. In a hospital context, for example, a '*Single room*' zone type will be defined in which a '*Bed*' position can be assigned to a patient, as well as a '*Door*' position and a '*Toilet*' position which cannot be assigned to patients but which can be used to associate a nurse call device. Another type of zone, '*Double room*', could have a similar configuration but with two assignable positions '*Door-side bed*' and '*Window-side bed*'. These two types of position are structurally hierarchical and can possibly inherit common configurations.

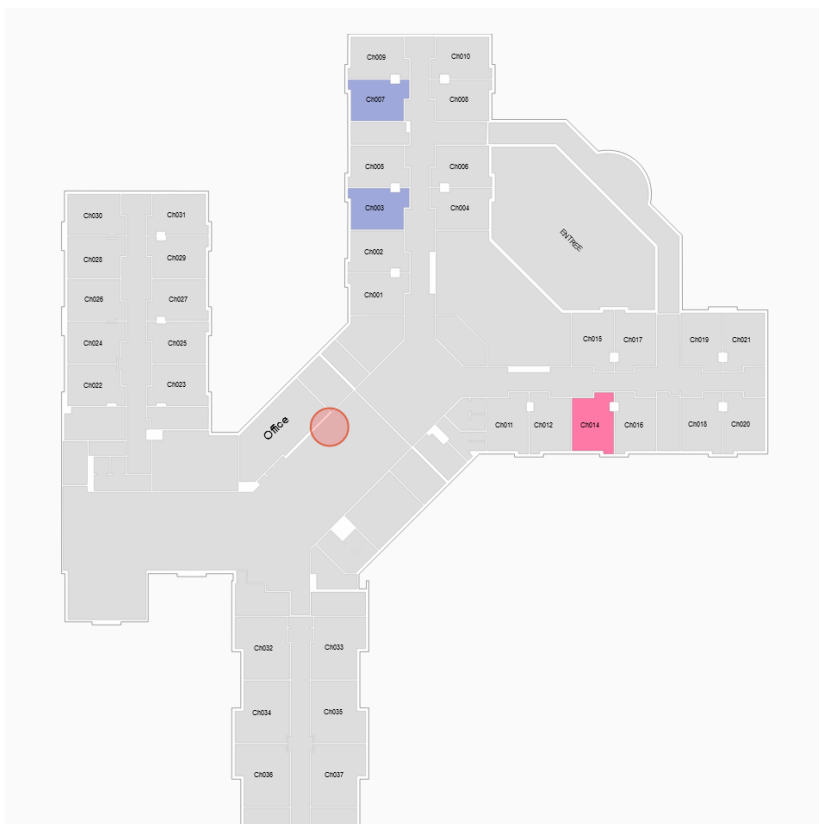
It is also possible to associate a '*tag*' with each type of zone, in the form of a keyword that can be used transversely in the writing of triggering rules (we can thus imagine associating a temporary tag '*in repair*' or '*Covid*', etc. with certain types of zones, in order to direct them towards specific scenarios when an event occurs in one of these zones).

Inheritance

As previously mentioned, the organizational structure of the institution is defined in Ofelia® in the form of a tree structure. As with event types, this approach allows for the inheritance of parent properties onto children. A given zone (for example a room) can inherit characteristics from its parents in the hierarchy (corridor, floor, building, etc.). This internal structure allows the requisition engine to be able to state precise rules according to the location of the triggering events and the presence or assignments of the personnel: it is indeed possible to state simple rules aiming at requisitioning the personnel of the floor in case of a call in a room, or to call all the staff present in a wing of the building following an alarm.

Floor maps

The software allows to visualize in real time on a map the events in progress and their evolution. These can be alarms (intrusion, fire, fall, etc.), missions, nurse calls or any supported event. This visualization is entirely based on the tree structure described above: this structure allows a relevant display whatever the current visualization level. During the initial configuration, the administrator associated a vector map (global ground plan, building, floor, corridor, etc.) with each level of the structure breakdown, on which the precise zones of each level (technical rooms, bedrooms, etc.) were placed, as well as the communication devices (nurse call boxes, runaway detection beacons, etc.).



When an incident occurs, the operator can spot the presence of an event on the map, and then zoom in the hierarchy to the level of the zone concerned. For example, a building map can be colored when an intrusion alarm occurs, allowing a guard to identify the floor, the corridor and the zone where the alarm occurred in order to proceed with further manual checks. A housekeeper in the hotel industry or a nurse on duty in the office can also follow in real time on the map of his floor or his department all the calls in progress.

Event monitoring and tracking

One of the purposes of the software, in addition to its smart requisitioning functions, is the automatic analysis and monitoring of system events. This can be a statistical or qualitative analysis in real time (monitoring the quality-of-service of each type of event according to specific criteria related to the duration of each transition), the definition of automatic visit reminders or the definition of "failsafe" rules allowing users to be alerted by specific scenarios if a critical event remains too long in a given state. The notion of priority, and the way in which it influences the call scenarios of users and their availability, completes this proactive management of the event life cycle. All these measures, plus the manual tasks and controls that can be defined for each event, guarantee a complete traceability of each alarm trigger and the possibility to retrospectively analyze the entire course of an intervention.

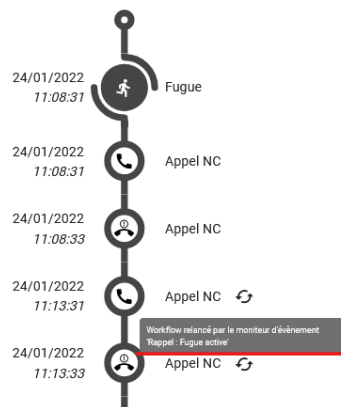
Event monitors

Event monitors are a set of items that allow to act on the life cycle of an event by reacting to transition times and defining appropriate actions. The inner rules of these monitors are defined globally in the system for a given type of event, then they are linked to specific types of events. Once associated, they are applied to each instance of that event type as well as to the children of that type, and are cumulative at each level: each instance of an event is thus monitored both by the monitors specific to its type, and by those defined for all its ancestors' types.

These event monitors are of three categories, each with distinct responsibilities:

Reminder Monitors

Defined for a given type of event, this type of monitor measures the time it takes to perform a given state transition (for example, the time that elapses between a nurse call and the actual presence of a nurse in the room): if this time exceeds a defined limit, the current transition state is forced again. This change of state then causes the cancellation of the current workflows and launches the rule engine to restart the most relevant scenario. This restart is accompanied by an increment of the workflow execution counter: this counter can be leveraged by the rules engine to launch an alternate scenario after several attempts for example.



Out-Of-Time Monitors

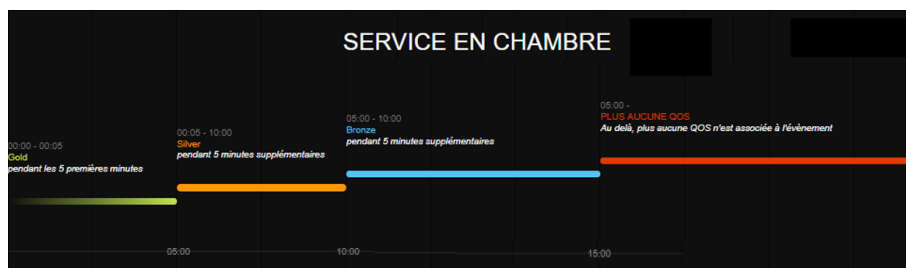
This type of monitor is specific to 'Mission' events. When a mission exceeds a certain time before transitioning to a given state (for example between its start and actual completion), the monitor allows one (or both) of the following actions to be performed:

1. Failing the mission (i.e., transitioning it into the failure state, which could in turn trigger another scenario)

2. Trigger another event whose nature and type are to be defined (it can be another mission such as an alarm, or any other nature)

Quality-of-service measurement

For a given event type, this monitor measures the *duration* of the specified state transitions. According to predefined duration ranges, a quality level is associated with the event. For example, it can be used to obtain an overview of the performance of interventions following a requisition made after a lone worker protection alarm: once the emergency personnel have been called and requisitioned, it is necessary to know whether the time taken to complete the intervention meets the internal service quality standards.



Nom : Temps d'intervention

Nature de l'évènement : Alarme DATI

Type de déclenchement du moniteur d'évènement : Suite à une réquisition réussie

Au changement d'état de l'évènement :

Depuis	Vers
Arrachement	Retombée
Perte de verticalité	
Immobilité	
Alarme Temporelle 1	
Alarme Temporelle 2	

Niveau 1 : Correct (Inférieur à 15:00)

Niveau 2 : A améliorer (De 15:00 à 25:00)

Dernier niveau : Trop long (Supérieur à 25:00)

These quality-of-service levels are displayed in real time in the operators' dashboards (one per type of event) so that they have immediate feedback of the events that have crossed the acceptability thresholds. These same service levels are also available afterwards in the form of statistics.

Visit reminders

Callbacks are a special monitoring of the '*nurse call*' event cycle that responds to a frequent scenario in hospitals: when an agitated patient makes abusive use of the nurse call system, the nursing staff has sometimes no choice but temporarily deactivating the switch. In order to prevent this patient from being "forgotten" because his or her device is disabled, a visit reminder can be used to initiate a time delay after which the room will automatically switch to the '*Visit Due*' state and trigger a call.

Unlike event monitors, a visit reminder is not linked to a particular type of event but is associated with a department (or medical unit): it applies to all nurse calls in the zones associated with the department and thus makes it possible to define a mandatory visit policy at regular intervals for the entire department. An option even allows to activate the visit reminder automatically whenever a nurse call switch is disabled.

Event Priorities

Not all events have the same urgency, which is why the software allows to define a priority level for a given type of event. This priority level can evolve during the life of the event and be used by the rules engine to trigger different scenarios depending on the urgency associated with a given event instance.

Priority groups

The priority levels do not necessarily have the same meaning or the same granularity depending on the nature or the type of event. You may want to define '*High*', '*Medium*' and '*Low*' priorities for maintenance missions, for example, and manage '*Urgent*' or '*Evacuation*' priority levels for fire alarms. In order to manage this coherence and to allow a definition as close as possible to the business needs, the administrator defines priority groups with the necessary levels. These priority groups are then associated to each type of event and inherited or modified on their children.

Change of priority

Changing the priority of an event is subject to a specific permission. Changing the priority level of an event triggers:

- Analysis of the rules engine to see if a more relevant scenario applies to this new priority
- Possible changes in the format of the messages sent according to the priority

Actions and Forms

The software allows to associate actions, in the form of entry forms, to an event. These actions are defined in templates (called *action templates*). These templates are used to create lists of actions to be carried out depending on each type of event and departments. Their usage is polymorphous and can be similar to maintenance schedules in industry (for missions) or biological value acquisition for care management.

These templates (each being a set of unitary tasks) are then associated with a department of the organization or overloaded for each type of event (and inherited substitutively on each of its children). The actions present in a template definition are unitary elements with a title, a flag indicating their optional or mandatory status, possibly a type of input value, and a return format.

The simplest action:

- Is optional,
- Contains a simple title

- Does not have a notion of data entry

=> it allows to indicate to a user a procedure to follow when an event of a certain type transits to a given state.

A more complex action can, for example, include a numerical value to be entered (e.g., temperature), and be mandatory: in this case, the state machine will be blocked until the action is completed. This can, for example, prevent the transition from a nurse call to an end of presence as long as a vital constant of the patient has not been filled in. The actions, as well as their associated forms, therefore allow the exchange of information (simple display of instructions or data collection) with the user each time an event reaches a given state.

Security

Licenses

Each deployment of Ofelia® software for a particular customer, whether it is a single or multi-tenant installation (see Supporting multiple tenants on page 77) results in the generation of a license valid for a given period.

The license conditions both the features available in the software and some of the volumetric values of these options (number of media per user, number of voice channels, etc.). For example, a site may be allowed to use the features related to lone worker protection, but be limited to a maximum of 5 protected users. New options can be unlocked or the number of authorizations changed by simply loading a new license into the software. This license can be in the form of a physical device (*ModuleKey*®) or as a file loaded by the administrator.

Licensing options

The following options can be activated in the license depending on the commercial contract signed by the customer:

Option	Type
Requisition module	Y/N
Number of callable users	Value
Statistics Module	Y/N
Number of LWP Alarms	Value
Number of Nurse Calls	Value
Number of technical alarms	Value
Number of voice channels	Value
Missions Module	Y/N
Mobilization Campaigns Module	Y/N
Number of Mobilizations/Notifications	Value
Number of input drivers	Value
ESPA output support	Y/N
External location providers (e.g., Stanley®)	Y/N

In the case of a *multi-tenant* deployment, the license needs to be configured for each individual tenant.

Connection

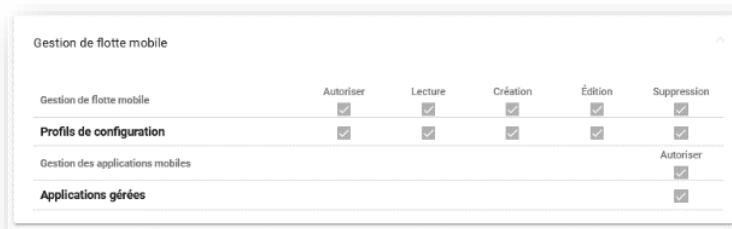
The connection to the administration and operation web interface requires a login/password pair specific to each user of the system. Ofelia® provides its own authentication management system, but it is possible to use additional identity providers: in this case, user authentication is delegated to a third-party system. This can be a provider based on a LDAP directory or a single sign-on (SSO) based on a *Microsoft Active Directory*® organization. Each identity provider declared in the system must be specifically configured by the administrator and has its own constraints and specificities.

Permissions

All the functionalities accessible in the software, whether for administration or operation, are subject to permissions that are evaluated according to the user who initiated the action.

Granularity

These permissions have a variable granularity depending on their scope: for example, to authorize access to a feature from the interface, the permission is a simple Boolean value (Yes/No), whereas for the configuration of a list, such as that of users for example, the permissions will be more complex (of the Read/Create/Modify or Delete type). Other elements can also benefit from dynamic permissions depending on the user's membership in a department or his relationship with the item being manipulated (the right to delete only the elements he has created himself, for example).

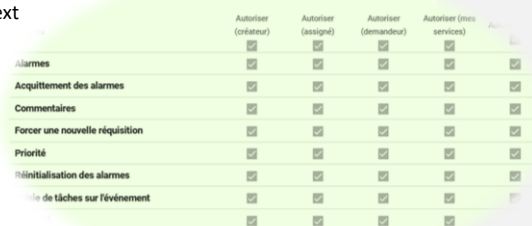


Gestion de flotte mobile	Autoriser	Lecture	Création	Édition	Suppression
Gestion de flotte mobile	☒	☒	☒	☒	☒
Profils de configuration	☒	☒	☒	☒	☒
Gestion des applications mobiles					Autoriser ☒
Applications gérées					☒

Dynamic permissions

Dynamic permissions apply to a specific number of actions in the system and are evaluated based on a complete context of the user and the event being accessed. These permissions involve the following relationships between the user requesting the action and the event on which the permission is based:

- **Creator:** the user is the originator of the initial generation of the event in the system
- **Assigned:** the user is directly assigned to the event (following an automatic or manual request)
- **Applicant:** the user is the requester of the event, for example in the case of a mission



	Autoriser (créateur)	Autoriser (assigné)	Autoriser (demandeur)	Autoriser (mess services)	Act
Alarmes	☒	☒	☒	☒	☒
Acquittement des alarmes	☒	☒	☒	☒	☒
Commentaires	☒	☒	☒	☒	☒
Forcer une nouvelle réquisition	☒	☒	☒	☒	☒
Priorité	☒	☒	☒	☒	☒
Réinitialisation des alarmes	☒	☒	☒	☒	☒
Plan de tâches sur l'événement	☒	☒	☒	☒	☒

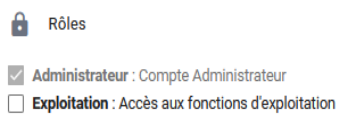
- **My Units:** the event takes place in one of the user's home departments. It is the event area that must itself be assigned to one of the user's units.

Roles

In order to avoid a tedious definition of each and every permission for each known user in the system, the configuration of a complete set of permissions is grouped in the form of a **role**. This role has a name and therefore defines a specific behavior of the software, e.g., for an administrator role, an operator role, an external staff role, etc.

Each user can then be assigned to one or more roles: the set of rights of each of these roles is then cumulated to determine what he can or cannot do in the product.

At startup and after each update, the application makes sure that the predefined '*Administrators*' role is always active and that it has all existing permissions.



System permissions

The different permissions that can be set in each role of the software are detailed in the following table, with the granularity/scope of rights that can be defined on each of them:

Category	Permission	Description	Scope
Alarms	Alarms	Access to this nature of events	Creator/Assigned /Requester/ My Units/ All
	Alarms acknowledgement	Transition to <i>Acknowledge</i> state	
	Comments	Adding comment to alarms	
	Force new requisition	Run again last requisition workflow	
	Priority	Change event's priority	
	Reset	Transition to Reset state	
	Tasks entry	Custom tasks and Forms	
	Visibility	Show/Hide the event in lists	
	Alarms Inhibition	Inhibit/Activate generators	Yes / No
LWP Alarms	LWP Alarms	Access to this nature of events	Creator/Assigned /Requester/ My Units/ All
	Alarms acknowledgement	Transition to <i>Acknowledge</i> state	
	Comments	Adding comment to alarms	
	Force new requisition	Run again last requisition workflow	
	Priority	Change event's priority	
	Reset	Transition to Reset state	
	Tasks entry	Custom tasks and Forms	
	Visibility	Show/Hide the event in lists	
	Alarms Inhibition	Inhibit/Activate generators	Yes / No
Nurse Calls	Nurse Calls	Access to this nature of events	Assigned /My Services/ All
	Cancellation	Manual cancellation of a nurse call	
	Comments	Adding comment to events	
	Force new requisition	Run again last requisition workflow	
	Priority	Change event's priority	
	Visit Reminder	Switch on/off Visit reminder on a nurse call box	
	Tasks entry	Custom tasks and Forms	
	Visibility	Show/Hide the event in lists	
	Nurse Call box Inhibition	Inhibit/Activate generators	Yes /No
	Update state value	Manual update of event	
System events	System events	Access to this nature of events	Assigned /My Services/ All
	Comments	Adding comment to events	
	Force new requisition	Run again last requisition workflow	
	Priority	Change event's priority	
	Reset	Manual Reset	
	Tasks entry	Custom Tasks and Forms	
	Visibility	Show/Hide the event in lists	
	Inhibition	Inhibit/Activate generators	Yes/ No
	Runaway	Access to this nature of events	
	Comments	Adding comment to alarms	
	End of runaway	Ability to terminate a runaway event	

Runaways	Force new requisition	Run again last requisition workflow	Creator/Assigned /Requester/ My Units/ All
	Priority	Change event's priority	
	Tasks entry	Custom tasks and Forms	
	Visibility	Show/Hide the event in lists	
	Sensor Inhibition	Inhibit/Activate runaway detectors	Yes/ No
Missions	Missions	Access to this nature of events	Creator/Assigned /Requester/ My Units/ All
	Assign mission	Manual assignation of a mission to a user	
	Comments	Adding comment to events	
	Quality Check	Update quality check flag on the mission	
	Start	Manual transition to 'Started' state	
	Fail	Manual transition to 'Failed' state	
	Pause	Suspend a mission	
	Priority	Change event's priority	
	Resume	Resume a suspended mission	
	Manual Requisition	Trigger a manual requisition	
	Tasks entry	Custom tasks and Forms	
	Complete	Manual transition to 'Completed' state	
	Visibility	Show/Hide the event in lists	
Mobilizations	Campaigns	Access to this nature of events	Creator/Assigned / All
	Mobilizations	Access to this nature of events	
	Stop a campaign	Manual transition to end state	
	Start a mobilization	Manual transition to 'Started' state	
	Handrail	Adding comment to a campaign	
	Priority	Change event's priority	
	Tasks entry	Custom tasks and Forms	
	Visibility	Show/Hide the event in lists	
	Create	Manual creation of a campaign	Yes/ No
	Start a mobilization	Start a mobilization as part of a campaign	
System Administration	Purge scheduling		Authorize/ Read/Create /Edit/ Delete
	Backup scheduling		
	Data Purge	Manual deletion of old data	
	Backups	Handling database backups	
	Configuration export	Download configuration (to upload it on another Ofelia)	Yes / No
	Close browser	Close Ofelia dedicated browser application	
	Import configuration	Delete all data and replace it with a saved configuration	
	Restore backup	Replace data with a backup	
	Download a backup	Download a backup file	
Temporal Alarms	Create	Create a new temporal alarm	Creator/Assigned / All
	Disarm	Disarm an on-going alarm	
	Browse	List temporal alarms	
	Extend	Extend an on-going alarm	
Configuration	Roles		
	Alarm Types		
	LWP Alarm Types		
	Nurse Call Types		
	System Event Types		
	Campaign Types		
	Runaway Types		
	Mission Types		

Mobilization Types		Authorize/ Read/Create /Edit/ Delete
Position Types		
Zone Types		
Location Beacons		
Nurse Call devices		
Mobilization buttons	Management of buttons that can be placed on a mobilization dashboard	
Calendars and Plannings		
External Applications		
Capability Converters		
Drivers		
Functions / Job roles		
Login IDs		
Action lists		
Medias		
Media Model		
Call Profiles		
Sounds		
Call Strategies		
Media Templates		
Message types		
Webhooks		
Event State definitions	Editing priority and state representation (color, caption)	
Files		
Technical Alarm Generators		
Mobilization Campaign Generators		
Mobilization Generators		
Technical support contact information	Editing contact information (not implemented yet)	
Medallion		
Unit regulation mode	Switch between manual and automatic unit regulation	
Zone regulation mode	Switch between manual and automatic unit regulation	
Campaign Dashboards		
User availability levels		
Staff /Employees	Edit staff members	
Technical Points		
Visit reminders policy		
Priorities		
Temporal alarm profiles		
Availability profiles		
User profiles		
Variables		

Scheduled Events	Scheduled Missions		Authorize/ Read/ Edit/ Create/ Delete
Exploitation	On-going Missions		Yes/ No
	Service view		
	Administration		
	Technical alarms		
	Temporal alarms		
	Workflow cancellation		
	Mobilization Campaigns		
	Customers		
	Mission creation		
	User profile edition		
	Events		
	History		
	Patients		
	Employees		
	Floor Plans		
	Statistics		
Mobile Devices Management	Configuration profiles		Authorize/ Read/ Edit/ Create/ Delete
	Managed Applications		Yes/ No
Healthcare	Nurse calls assignation	Manual assignation of nurse calls	Authorize/ Read/ Edit/ Create/ Delete
	Patients	Patient list	
	Presence type	Presence type management (nurse, doctor, etc.)	
Hotel	Customers	Customer list	Authorize/ Read/ Edit/ Create/ Delete
	Loyalty programs		
Event Monitors	Out-of-time		Authorize/ Read/ Edit/ Create/ Delete/ Read All
	QoS	Quality of service	
	Recalls	Recall Monitors	
Business Rules	Keywords	Tags management	Read/ Create/ Edit/ Delete
	Trigger Rules	Managing workflow triggering rules	
	Workflows	Workflow definition creation	
Security	Roles	User roles management	Yes/ No
	Update license	Change software license level and features	
	Security	Permission configuration	
	Reset password	Reset passwords (not only connected user's)	
Statistics	Dashboards	Edit Dashboards	Authorize/ Read/ Edit/ Create/ Delete
	Time slots	Custom (user defined) time slots management	
Zones and Units	Units	Managing Services (Functional units)	Authorize/ Read/ Edit/ Create/ Delete
	Zones	Managing facility areas (rooms, corridors, etc.)	

Lone Worker Protection



A major functional aspect of Ofelia's software is its ability to manage lone worker protection devices (aka LWP, aka ITP – Isolated Worker Protection - or IWAP – Isolated Worker Assistance Devices). The purpose of this feature is to protect the company's employees from potential risks of aggression, falls, faints or accidents. In a context where employees are increasingly exposed to working in isolation, the software relies on a number of physical devices to ensure proactive monitoring and geolocation of staff both indoors and outdoors.

Detection

The system provides drivers to interface with ASCOM® and Stanley® LWP solutions to detect the user's location and the status of his alarms. These alarms can be triggered voluntarily (voluntary alarms, test alarms, cord pull) or automatically (loss of verticality, immobility).

Temporal alarms

In addition to the detections directly linked to the sensors of the mobile devices, it is possible to set up alarms that are triggered automatically after a timer has elapsed. These particular alarms allow to limit the hazards during two frequent situations to which workers are exposed:

- **Journeys/Routes:** By arming a time alarm proportional to a journey time (or, indirectly, proportional to a distance measured in number of steps for example), the worker indicates that if the alarm has not been deactivated at the end of his journey it should be triggered to warn of a potential accident. When the waiting time expires, the user is warned of the impending triggering: he can then either extend the duration of the trip by a predefined number of units or deactivate the alarm altogether to indicate that he has reached his destination. If neither of these two options is taken, the software detects a failure and can trigger the emergency scenarios provided for this purpose.
- **White zones:** In a similar way to the routes, a user in an area not covered by detection and communication systems (isolated environment, elevator shaft, etc.) can indicate that he will not be reachable for a defined period. At the end of this period, he is warned of the imminent triggering of the emergency alarm: he must then leave the white zone to extend or end the intervention. Failure to do so will result in an alarm being triggered and the associated requisitions in the software to start running their workflow.

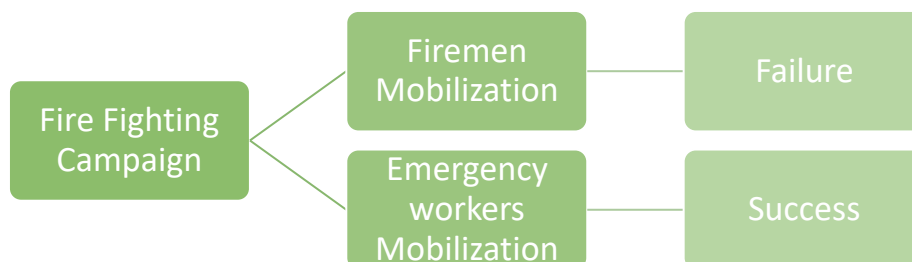
Mobilization Campaigns



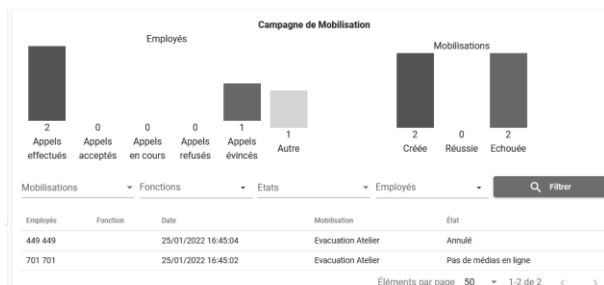
The mobilization campaign management module is designed to respond to the problem of mass notification of a large number of personnel: for example, this may involve emergency plans for hospitals (attack or health emergency plans) or evacuations from Seveso-Category sites. In the context of these campaigns, the objective is twofold: on the one hand, to provide a simple, "push-button" means of sending out massive predefined notifications, and on the other hand, to allow real-time monitoring of the progress of the calls (number of people called, number of messages transmitted, user acknowledgements, etc.). These two aspects are managed separately in the application: creation by an administrator of dedicated screens for operators (button layout with predefined scenarios), and real-time use and monitoring by operational staff.

Campaigns vs Mobilizations

Ofelia's mobilization campaigns are actually based on two types of events that share a hierarchical relationship: a *Mobilization Campaign* event groups together a set of *Mobilization* events children. The campaign has a simple life cycle (*Started/Stopped*) and each of its Mobilizations children has its own state cycle depending on its individual success (*Start/Fail/Success/Cancel*).



When using a campaign, an operator is provided with a real time summary of all calls made for each of the mobilizations that compose the campaign. This allows him to follow the complete evolution of the event without the need to browse each of the sub-elements. He is directly presented with aggregated information such as the number of calls in progress, the failures, etc.



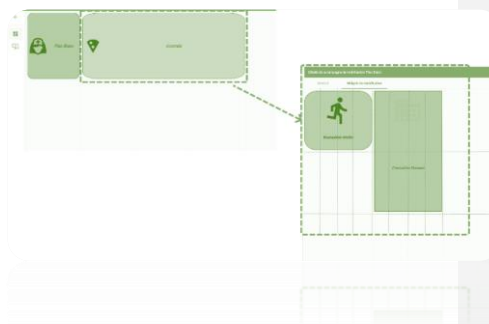
Campaigns Templates

Each mobilization campaign is associated with a type of campaign event to generate as well as an operational screen. This screen is made up of a 6-column wide grid on which the administrator can place trigger buttons of different sizes and rendering (color, icon, text, rounded edges, etc.). Each of these buttons is associated with a type of mobilization event to be created and a workflow to be triggered.

These buttons can be defined in automatic start (in which case the associated mobilization will start at the same time as the parent campaign) or in manual start, i.e. when an operator presses the button.

Campaigns dashboard

Once the different campaign models have been defined, the administrator groups them together in a dashboard. This is a new level of graphical grouping, still in the form of a customizable grid on which campaign templates rather than mobilization buttons are placed. The dashboard is unique within the application (per tenant in the case of a multi-tenant deployment). Pressing one of these templates will trigger the corresponding campaign template, giving the operator access to the buttons it contains. The goal is to provide a simple way to access all campaign triggers for specific operator groups. By adjusting proper permissions, an operator in a control room can for example have limited access only to the triggering dashboard and to the monitoring of the campaigns in progress.



Operational follow-up

Once the templates and dashboards are configured, an authorized operator can launch campaigns and follow their progress. He can start or stop a campaign from the dashboard, and then within each campaign trigger manual mobilizations by pressing the buttons pre-programmed by the administrator. A summary of the calls allows him to follow in real time the progress of the whole campaign.

Event Generators

Event generators are a central element of event management. Indeed, only events of certain natures are created manually by a software user, all the others are produced automatically by a generator. A generator is a software element that uses different elements of the execution context to decide on the creation or evolution of an event in the system: for example, for nurse calls, a dedicated generator is able to use the knowledge of other calls in progress in the room, information on inhibition or control of the boxes or other data to decide whether or not to create a new event and to choose its type and initial state.

Some generators are internal and have their own business rules for managing events (for example, nurse call generators), while others are accessible to the administrator, who can customize the rules for handling events according to the values of dynamic variables: these are the alarm, mobilization and campaign generators.

Variables

All drivers with the 'Variable' capability are able to communicate with Ofelia by updating a variable defined in the system. Similarly, Ofelia[®] workflows can update a variable value and, if the driver supports it, the information can even be sent to the remote device connected to the driver (PLC, etc.). This is for example the case of the Modbus[®] driver, which allows bidirectional communication with Ofelia[®].

A variable is a simple named element linked to a driver instance and to an address within the external system managed by this driver. This address is an alphanumeric string whose meaning is specific to each driver. It should be noted that, although it is possible to write in all the variables defined in the system, not all the drivers allow a bidirectional communication and the writing of a variable in a workflow will thus not always be sent to the remote PLC.

Alarm generators

An alarm generator is a set of scripts linked to a zone which allows to match rules on variable values with the creation or evolution of an alarm event in the zone. For each state of the alarm (active states [*normal*, *low*, *very low*, *high*, *very high*], *acknowledgement*, *reset*), the administrator can associate a Boolean expression involving one or more variables linked by logical conditions. It is also possible to define a duration during which this expression must be checked before applying the state.

For example, the administrator can decide that an alarm must be activated when the variable NUUTX_DM is 1 and the variable NUUTXTEST is greater than 6 for more than 5 seconds:

Alarme (normale)

Script *

@NUUTX_DM==1 && @NUUTXTEST>6

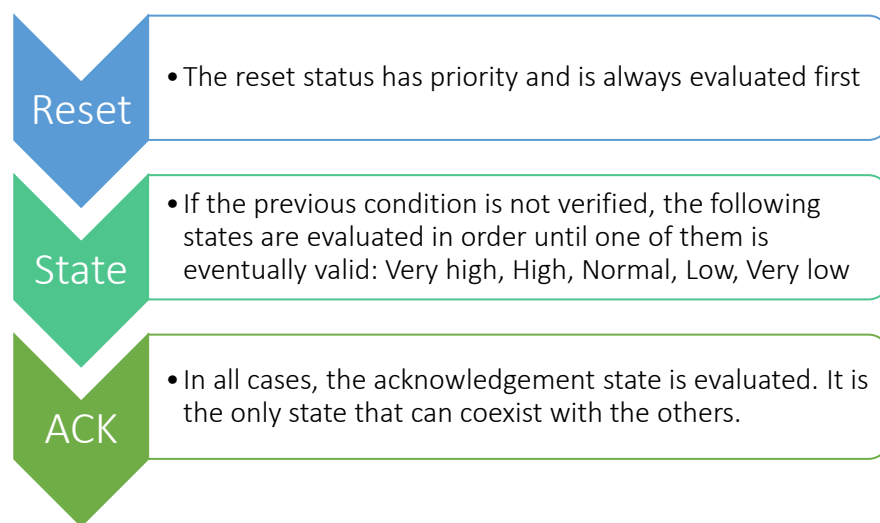
Durée pendant laquelle le script doit être valide (en sec)

5

An alarm generator manages a single event and carries the logic of its state machine: when the values of the variables change, the generator takes care of creating a new event in the zone if there is none active or of changing its state if an instance is already in progress. An alarm generator is thus parameterized with the type of target alarm event and possibly all the parameters necessary to be autonomous in the creation of events.

Precedence of states

Since there are multiple alarm states, several Boolean conditions for changing state can be checked simultaneously, for example if a state requires the condition *Variable* ==2 and another *Variable* <=5. To evaluate the state to assign to the event, the generator applies the following rules:



Thus, if the 'High' state is checked as well as the 'Acknowledge' state, then a 'High' value alarm is created and immediately acknowledged.

Mobilization generators

The purpose of a mobilization generator is to create mobilization events, or more precisely to create notifications. It works in a similar way to an alarm generator, but with a unique script because the Mobilization type has only one configurable state, the 'In progress' state. Such a generator allows to send simple notifications to users each time a set of conditions is met on variables. It can be used for example to forward a value acquired on a sensor to a group of users.

The rest of the characteristics are identical to all generators, namely: a zone, as well as an event type and its associated parameters. The particularity of a mobilization generator is its capacity to have several instances of events active simultaneously (contrary to the alarm generator, which evolves a single instance on the zone).

Mobilization campaign generators

The campaign generator is very similar to the mobilization generator. It allows to start not a single mobilization but rather a complete campaign following the evaluation of the value of one or more variables.

As with the mobilization generators, several simultaneous campaigns can be generated if the condition associated with the single '*In progress*' state is valid before the end of the previous campaign.

Files

Each configurable event generator (alarm, mobilization or campaign) allows you to associate a list of files with the generator. These files are then automatically attached to the generated events. In the case of voice calls, the sound files are also automatically played whenever the associated workflows are called.

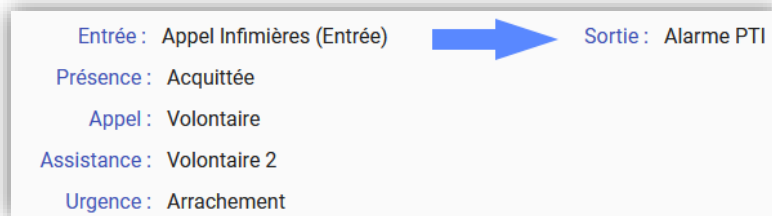
Capability converters

In addition to manual creation or the use of programmable event generators, another way to create events in Ofelia® is to convert a source event into another type of event. To do this, an administrator can define a capability converter to translate an event or data received on a driver into another event. Even more indirectly, a received data can be used to feed a variable which, in turn, will cause the creation of an event through a dedicated generator.

These converters belong to three different categories and their parameterization is more or less simple, sometimes using block programming or the use of regular expressions to analyze data transmitted to a driver in text form. Once set up, a converter is associated to a driver instance and it then governs all the event creations of this instance by translating the information received into a target event type.

Nurse call to LWP alarm

This type of converter is the simplest: it allows to associate each state of a *Nurse Call* event into a lone worker alarm state. A nurse call using this converter will then be automatically transformed into a LWP alarm event.



Text to Nurse Call Converter

This type of converter aims at exploiting a text frame (ASCII) received via a driver instance to transform it into a *Nurse Call* event. The objective is therefore to extract from the received frame the information necessary to create or evolve the nurse call event (room, current status, presence information, etc.).

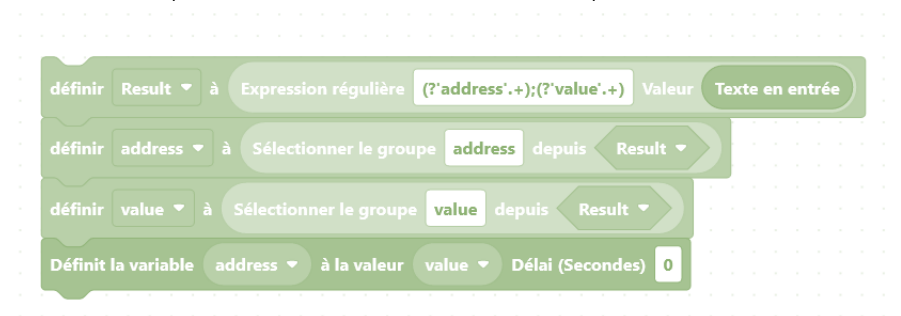
Since the format of the received frames depends on each third-party data provider, the system allows to define the way each frame should be split to extract the information specific to each manufacturer. The breakdown can be done using regular expressions, based on fields separated by a specific character

(semicolon for example) or an even more complex pattern. The expression of this analysis is done graphically by a programming system in the form of blocks.

Google's Blockly[®] visual programming system is enriched with blocks specific to Ofelia[®] that allow the manipulation of the input text and the extraction of the state and the area of the events produced. In order to facilitate the setup of the most common providers, a built-in customizable template is provided for semicolon-separated text frames, as well as for frames specific to the TELEVIC[®] and ACKERMAN[®] proprietary nurse call systems.

Text to Variable Converter

This type of converter also allows the analysis of an input text frame (ASCII) but instead of using the extracted elements to proceed directly to the creation of nurse call events, the logic implemented allows the manipulation of the value of one or several variables. Associated with event generators (see *Event Generators* on page 61), such a converter can, for example, trigger a mobilization or an alarm upon reception of a text frame. The following example shows the parsing by a regular expression of an input frame in order to update the value of a variable whose name is also passed in the frame:



In order to facilitate the configuration of the most common systems, a predefined and customizable template is provided for semicolon separated text frames, as well as for OAP[®] and ESPA[®] specific frames. These templates can be used as is or adapted to fit the format of any text frame received on a driver instance implementing a converter capability.

Statistics

The exploitation of the traceability data generated by the software is done through two distinct modules:

- **Custom analysis:** On-demand graphical generation of histograms and other visualizations representing a specific request (time range, type of events, etc.)
- **Dashboards:** Dashboards calculated at regular intervals, providing side-by-side temporal representations of various indicators (average call duration, number of alarms, etc.)



Custom analysis

The request for a personalized analysis is made in three steps

- **Definition of the filter:** This involves defining the selection of events to be analyzed. The choices include:
 - The **nature** of the event (Mission, Alarm, Nurse call, etc.). This is the only mandatory parameter of the filter
 - The **date ranges**
 - The **types of events** concerned (multiple choices among the types of events of the selected nature). Note that inheritance applies and that the selection of a parent type automatically implies all its children.
 - **Functional Units** (multiple choices): these are the departments (medical units) to which the zones where the events occur are attached
 - The **type of zone** where the events occurred (for example, only events located in a room or in a corridor)
 - The **zones** (multiple choices): areas in which the event took place. It can be a final level (for example a room) or an intermediate level where all the descendants are grouped together (for example a floor where all the corridors and rooms are grouped together)
 - **Default functions** (multiple choices): this filter allows you to limit the selection to events for which one or more particular functions (jobs) are defined
 - **Applicant:** limits the selection of events to those requested by a particular employee.

1
Filtrer

Nature de l'évènement *
Appel Infirmière

Date de début
Date de fin

Types d'évènement

Services

Type de zone

Zones

Uniquement les évènements avec la fonction par défaut

Demandé par

Choice of analysis type: The statistics module allows to analyze and render three categories of indicators (PKI) based on the data filtered in the previous step:

1. The number of occurrences of the events
2. The value assigned to their quality-of-service (e.g., the proportion of assignments completed for each service quality level)

3. The duration of one of their transitions (e.g., the average time for a nurse call to go from the call state to the nurse presence state)

Choice of grouping and aggregation mode: This is to define the grouping according to which the elements filtered and selected in the previous steps will be displayed, as well as the aggregation to be applied (sum over the period, average, etc.). These parameters depend on the type of analysis.

Dashboards

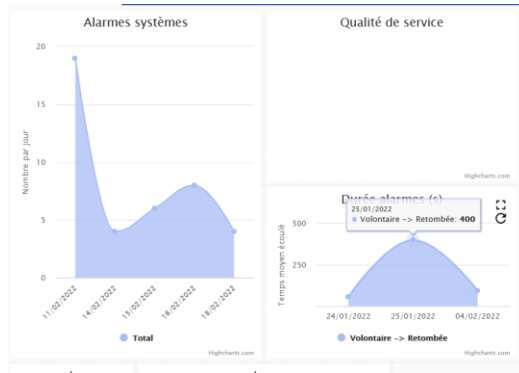
The use of instant analyses consists of defining a query and requesting its display in graphic form. Although this functionality allows to explore the system's data once in a while, it remains tedious when you want to follow the same indicators on a regular basis. To meet this specific need, Ofelia® has a statistics dashboard feature.

The Ofelia® administrator can create different dashboards, each one grouping several predefined graphs. A dashboard is made up of a variable-width grid (for example 2 or 4 columns) on which the administrator can place graphical 'widgets' occupying a predefined number of columns. This makes it possible to organize several representations simultaneously on the same page, while adapting to the user's screen.



Widgets

Each of the graphical widgets is set up in a way similar to a custom analysis (see on page 65).



Only the notion of date is specific: while it is indeed possible to define a fixed date range (just like for a custom analysis), the widgets also support a much more powerful mode in the form of *dynamic date range*. These ranges allow the definition of intervals such as "current week", "previous quarter", "last 5 days", etc. that would be impossible to define using fixed ranges.

The process of defining an interval of dynamic dates consists in fixing the unit of measurement (days, weeks, months, etc.) and in expressing how much one goes back in time from the current date, as well as the range duration.

The following definition can be used to display the **current week**:

Unité de temps de départ *	Décalage par rapport à la date courante *	Unité de temps de l'intervalle *	Durée *
Semaine calendaire	1	Semaine	1

- From the date of execution of the report, go back one calendar week (i.e., positioning to the previous Monday) and then take a duration of one week long.

To select the **previous working week** from the current one, from Monday to Friday, the definition would change on the length of the interval:

Unité de temps de départ *	Décalage par rapport à la date courante *	Unité de temps de l'intervalle *	Durée *
Semaine calendaire	1	Journée	5

System Administration

Installation

Once the prerequisites are installed (mainly *Microsoft SQL Server*® and *RabbitMQ*), a step-by-step installation and configuration program allows you to deploy Ofelia in a single-server installation running *Microsoft Windows*®. Any other configuration (multi servers, clustering, non-Windows operating system, etc.) must be deployed and configured manually.

On the same environment (single *Microsoft Windows*® server), updates are also applied through an official installation program. This program updates the software and takes care of the possible migrations and evolutions of the underlying database. The version number of the application is semantic and adopts the following convention:



Backup and Restore

The software performs an automatic backup of the database during the version each upgrade following an update: if the update fails, the previous database is kept as a backup and restored automatically the next time the application is started.

In addition, the application has a schedule backup mode (every day at a fixed time) as well as an option to perform backups and restores manually.

Location of backups

If nothing is specified in the configuration file, the backups are saved by default in the `[ProgramData\Appliware\Backups]` directory of the server where Ofelia® is installed. In the case of a deployment where *Microsoft SQL Server*® is located on another server than the application server, it is necessary to indicate the location of the backups in the configuration file as a network share and to make sure that both Ofelia® and SQL Server® service accounts have proper write access permission to both the folder and the share.

Version compatibility

When restoring a backup, its version is compared to the version of the software being run: if the database is older than the software version, it is first updated and the data is migrated to the target structure upon the next restart. If, on the other hand, the database is more recent, the application refuses to start until it has been updated to support this version of the database, thus preventing any risk of data loss or corruption.

Scheduled Automatic Backups

The administrator can also schedule an automatic database backup: this will take place in the directory defined for all backups and the administrator is free to define the periodicity of this backup as well as the retention policy and the time at which it takes place (though it is limited to only one per day and per schedule). The following example shows the configuration of a weekday backup at 11pm, keeping only the 5 most recent files:

The screenshot shows a configuration window for a backup. On the left, under 'Nom *', the backup is named 'Sauvegarde quotidienne'. Below this, 'Nombre maximum de sauvegarde à conserver' is set to 5. The 'Heure de début' is set to 23:00. On the right, the frequency is set to 'TOUTES LES SEMAINES' (All weeks) with a dropdown arrow. Below this, it says 'Toutes les 1 semaines'. There are buttons for the days of the week: 'Lundi', 'Mardi', 'Mercredi', 'Jeudi', 'Vendredi' (all highlighted in blue), 'Samedi', and 'Dimanche' (both in white). At the bottom, 'À partir du' is set to 26/1/2022 with a calendar icon.

It should be noted that, although the execution of a backup can be done while the software is in use, the intensive usage of disk and processor resources can lead to significant slowdowns of the product. It is therefore recommended to schedule backups at times when the software is not in use.

Manual backups

At any time, the administrator can also, from the interface, request the creation of an instant backup of the database. The backup is saved in the directory defined for all the backups and the precautions concerning the processor and disk load imposed on the application apply as for an automatic backup: it is therefore important to make a backup only at a time of non-critical use of the software. As with scheduled backups, a dashboard allows an administrator to view the history of manual backups performed and to estimate their volume, to consult any error reports and to choose whether or not to erase the backup to recover disk space.

Manual restoration

From the list of backups, whether automatic or manual, the administrator can restore an old data set. This operation causes all users to log out and all background processing (including processing of current event workflows) to be suspended. Users connected to the Ofelia web application are redirected to the home page where they are prompted to wait for the software to come back online before reconnecting. The software is automatically reactivated after the restoration and any version upgrades of the restored data. However, users who were previously connected must reconnect.

Configuration Import and Export

In addition to backup and restore, which concern both configuration and operational data (event history, etc.), a built-in configuration import/export mechanism allows to target only the application's parameters.

This functionality allows the following use cases for example:

- A consultant can configure a complete environment for his client on his workstation and then provide it to him in the form of a single file so that he can initialize the system in production
- During a test phase a tester can extract a configuration from a production environment to provision a test environment
- At the end of a training phase, an administrator can reset the system by reinjecting the configuration before the final production launch

The configuration import function is done in the same way: the system downloads the file and analyzes its content to validate that the version of the configuration contained is compatible with the current application version, then the users are disconnected from the system, all existing data is deleted and the new configuration replaces the current one. At the end of the import, users can log back into the system and start again on a completely blank base in terms of history.

Purge

Manual purge

In order to limit the volume of data stored by Ofelia® while respecting the traceability strategy defined in the facility, the administrator can ask the software to delete all data concerning events that took place more than a certain number of days ago (for example all event data older than 3 months).

Scheduled purge

To automate the cleanup process, the administrator can define a data purge policy. This policy, unique for each installation (or each tenant in a multi-tenant deployment), allows the administrator to choose the maximum age of the events and the time of the daily purge execution. At the scheduled time (preferably outside of the application's intensive use hours and after the end time of any scheduled backups), all events older than the age limit are deleted from the database.

The software provides a history of all manual and scheduled purges that have taken place, along with a summary of the number of days of retention requested and the status of the operation (*in progress*, *completed*, *failed* with any associated error message).

Exploitation

Filtering

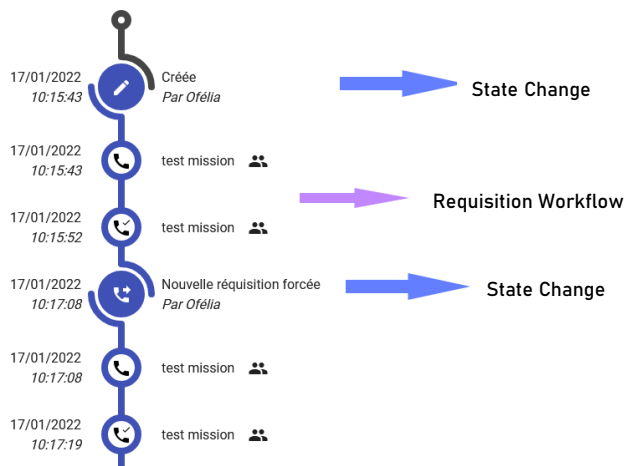
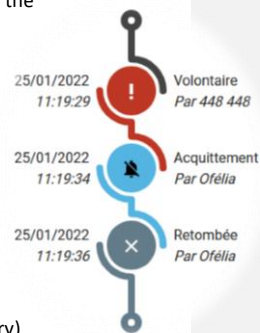
The management of user rights coupled with the functional division of licenses and features allow each operator of the application to be presented with a web application restricted to only those functions that he has the right to manipulate. Furthermore, the assignments to the zones and departments of the facility further refine this filtering by offering the user a vision centered on the floor or the unit(s) he is in charge of.

An event-centric approach

As detailed above, most of the elements handled by Ofelia® are events of various kinds: whether they are system faults, technical alarms, runaways, mobilizations or missions, all these entities can be handled in a coherent and homogeneous way. Their representation is therefore similar, and only the properties specific to each type are displayed separately. Hence it is very easy for the user to understand the various functional facets of the product because the knowledge about the functioning of one type is immediately transposable to other types of events.

Timeline

Navigation among the events is classically performed through a set of dashboards and detail screens. The data on the screen is updated in real time, making it easy to follow in real time all the events occurring on a floor plan, the representation of the beds of an entire department, or the progress of each stage of a workflow or the changes in an event status. In the latter case, each event (either in progress or in history) has a timeline that allows to follow each step of its evolution, with a time stamp and the associated actions (scenario launch, requisitioned personnel, etc.).



Dashboard Lists

The operator can access summary views of the current and past events allowing him to visualize the activity and to choose the event for which he wishes more detailed information. These views are generic and filterable, displaying by default all the events based on their common data (nature and type, name, zone, date, requisitioned personnel, etc.). Specific screens more focused on missions (immediate or scheduled) or nurse calls are also provided to address specific requirements.

Events in progress

The list of events in progress on the system presents all the events currently running, displaying by default the most recent ones at the top in order to provide an operator with a synthetic and instantaneous vision of the activity. Like the other views of the interface, the rows and columns are updated in real time as each event evolves. A supervisor can therefore use this single screen to monitor alarms and the progress of missions. The elements displayed in this summary view are the following:

Evénements											
Evénements à venir			Evénements des 8 dernières heures								
	Icone	Date de création	Durée	Type d'événement	Etat	Contrôle qualité	Code	Nom	Zone	Type de position	Assigné à
11		28/05/2022 09:52:29	41 mins	Driver Instance défaut système	Actif		S_927F	#PORTUABLE(DORE) Achermann (systeme) Driver Instance défaut système	Etablissement		
11		28/05/2022 09:52:19	41 mins	Driver Instance défaut système	Actif		S_927C	#PORTUABLE(DORE) Chai (systeme) Driver Instance défaut système	Etablissement		
12		28/05/2022 16:43:25	13 17h absence	Campagne de Mobilisation	En cours		MIL_0548	Campagne de Mobilisation	Etablissement		Offila
12		28/05/2022 16:43:47	13 17h absence	Campagne de Mobilisation	En cours		MIL_0548	Campagne de Mobilisation	Etablissement		Offila
13		11/05/2022 10:13:43	13 18h 13 mins	Missions	parti		M_2788	Missions	Etablissement	445 445	Offila

Column	Details
 Nature	Nature of the event (Mission, alarm, etc.) represented by its icon
 Priority	Event priority level (see Event Priorities on page 49)
 Out-of-time	<i>Specific to Missions.</i> Icon showing if the event has reached the time limit set in an Out-Of-Time monitor (see Out-Of-Time Monitors on page 47)
Creation date	Date and time (to the second) when the event was created
Duration	Current duration (to the minute) since creation
Event type	Event Type (with its associated color)
State	Current state of the event
Quality check	<i>Specific to Missions.</i> Flag showing the user has requested a quality check before its termination.
Code	Unique auto-generated code identifying the event
Name	Event caption, generated according to a formatting rule defined by the administrator for the event type
Zone	Name of the zone where the event occurred. It could be either creation area (for a nurse call), target zone (for a mission) or even current zone (for a runaway): the most appropriate zone is chosen by the engine based on the event's nature.
Position type	Event's position in the zone itself, for instance 'Bed side', 'Door side' or 'Bathroom' for a Nurse Call.
Assigned to	Coma separated list of all staff members requisitioned for the event. The list gets enriched as more and more employees accept the call.
Requested for	People benefiting from the event. For instance, for a mission, it could be the room's occupant.
Created By	Application user who triggered the event creation. If the event was auto-generated by a rule or a workflow, this field is empty.
Comments	Text comment associated with the event, for instance, an explanatory note added to a mission by its creator

Pagination and Filter

The events displayed in the dashboard are paginated (with a customizable page size per site and per user), and can be sorted by most of the relevant columns of the table. The list presented can be filtered by many search criteria:

Criteria	Description
Code	Exact code of the event
From	Start of a date range (end of range is optional)
To	End of date range (start of range is optional)
Event Natures	Event natures List (Mission, Alarm, etc.)
Event Types	Event types list. Type hierarchy applies: filtering on a parent type will include all events of the selected type AND of its children's types
State	Current state of the event
Quality of service	Only select events complying with a given QoS
QoS Levels	QoS levels list belonging to the aforementioned QoS: only events which have reached these levels will be returned
Priorities	List of priorities the events should have in order to be selected. Only current event priority is concerned.
Zones	List of zones where the event is currently in progress
Services	List of services of the event zones: only the events taking place in the zones belonging to these services will be returned

Quick actions

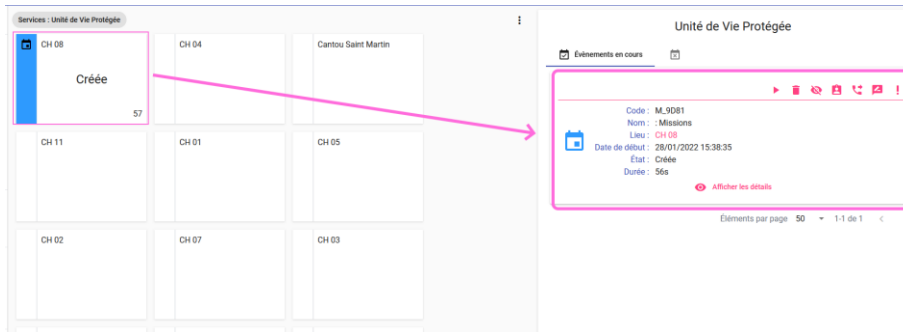
The dashboard allows access to quick actions without having to open the event details first. These are accessible via a shortcut on each event line and allow an operator to perform operations depending on the context and his own dynamic permissions in the system: modification of priority, change of status, manual requisition, addition of a comment, etc.

History

The event history dashboard is similar in all respects to the current event dashboard, except for the following features:

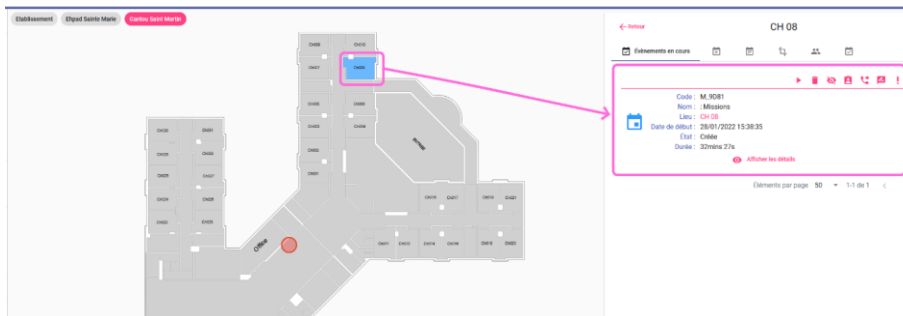
- Only *completed* events are displayed
- The status displayed is not the *current* status but rather the *initial* status of the event (when it was created)
- The filter on the status also concerns the initial status
- The list of events can be exported as a PDF or *Microsoft Excel*® file
- No quick action is available on the displayed events (because they are finished)

This display setting is better suited for monitoring by permanent operators (or nurses in a ward for example) than a simple summary dashboard. The relevant information is highlighted directly on each box representing a room, indicating the current status and giving access to the event details as well as to the current actions that the user has the right to perform on the events.



Floor maps

If the administrator has properly associated the various areas of the facility on hierarchical plans (one per building, floor, corridor, for example), events can be displayed in real time on the map by an operator. It is thus easy to locate alarms and other events in real time centrally by associating them with their geographical location.



Users

The list of users found in the system (whether employees, patients or customers) is created by the administrator or synchronized by gateways, but the associated information must be available for consultation and even manipulation by authorized users, even in an operational context.

Patients

A simplified interface allows the list of patients to be consulted and their stays and movements within these stays to be managed simply. In the majority of cases, this management will be handled outside Ofelia® by the admissions software. However, in the context of a small retirement home or a smaller rehabilitation center, an authorized employee could be responsible for creating stays and patient movements in the system.

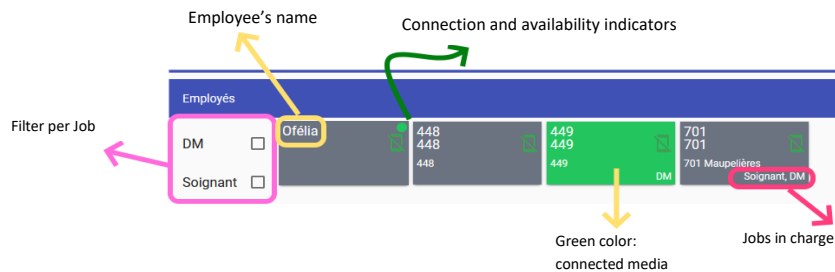
Clients

The management of hotel customers can also be carried out by an external system and then synchronized by a gateway. An Opera® compatible driver is available with the Hotel vertical license. In spite of this automation, a simple interface allows a minimal management of the customers, limited to their civil status, their language and an optional loyalty program.

Employees

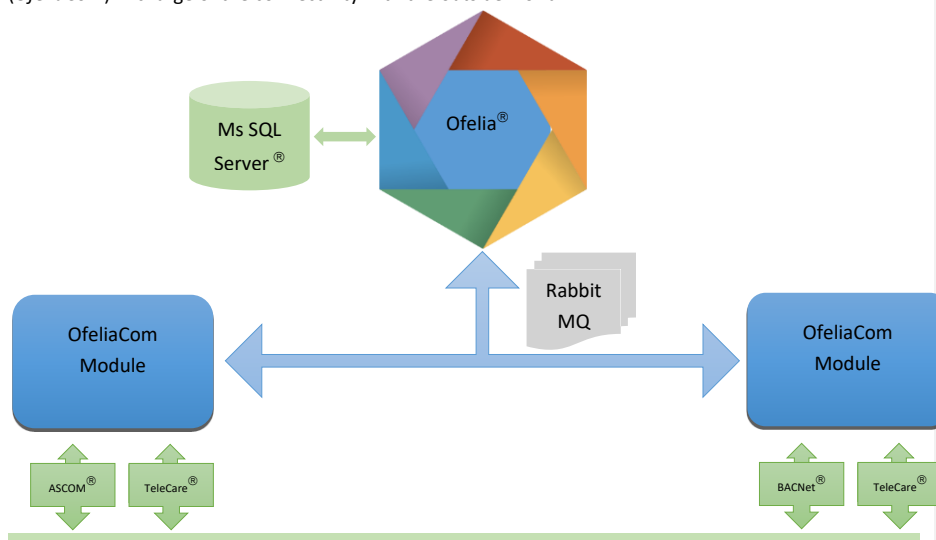
The employees are a central element of the application: they are the ones who are called and requisitioned in the different scenarios. An operator of the application must have a synthetic and relevant view of all the employees present in the whole facility, with information regarding their availability status, the function they occupy according to their schedule, etc.

A dedicated view allows a dispatcher to access all this data centrally, along with the real-time status of each staff member in a compact grid:



Deployment and Topology

Ofelia® is a server software based on other standard components on the market: *Microsoft SQL Server®* for its database, and *Rabbit MQ* for its asynchronous message bus. The entire application is modular and can host one or more main server nodes (cluster configuration) as well as several communication modules (*OfeliaCom*) in charge of the connectivity with the outside world.



Several instances of the same driver can be distributed on multiple *OfeliaCom* modules. Different topologies are available to distribute these components, depending on the expected load and volume of transactions as well as the hardware resources available on the site:

Mono Server

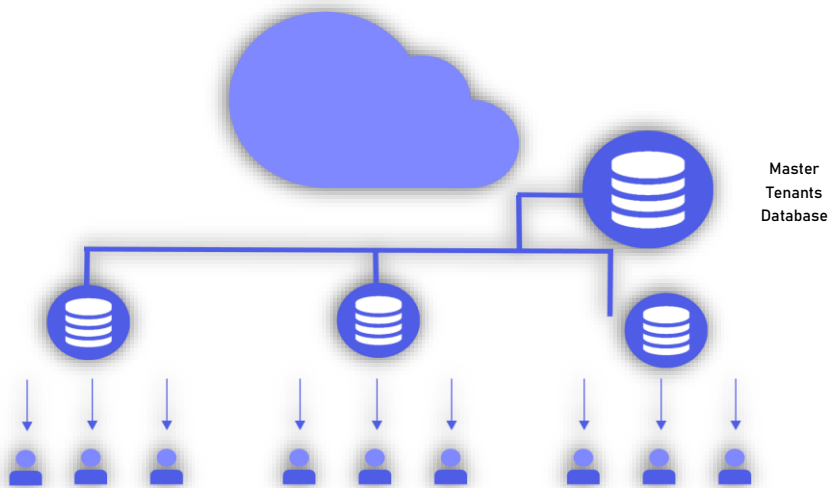
In its most common configuration, Ofelia® can be deployed on a single appliance providing web server, middleware, and database server functionalities. A *Microsoft SQL Server*® instance is first deployed on the machine, as well as a *Rabbit MQ* message broker. The installation of Ofelia® completes these prerequisites by adding the middleware layer and a graphical user interface (GUI).

Multi Servers

The communication between Ofelia® and the various driver instances can take place through *OfeliaCom* communication modules: these are independent deployments of a part of the software along with the driver concerned, communicating with the Ofelia® server(s) through *Rabbit MQ* message queues. The single-server mode described above is only a specific implementation of this native multi-server mode, in which both the Ofelia® core software and its communication modules are jointly deployed on the same machine. In the same way, the *Microsoft SQL Server*® database server can itself be located on a remote server (a cluster or an *Always-On high* availability group), it is sufficient that the necessary network connectivity is present between the different machines that constitute the deployment.

Supporting multiple tenants

Ofelia® software is designed from the start to allow multi-tenant hosting. The multi-tenant mode is therefore native even if a vast majority of the applications deployed are in single-tenant mode. The distribution of tenants is done by '*sharding*' the data, i.e., by distributing each tenant in its own isolated database. In the case of a multi-tenant deployment, there is a database for each tenant, as well as an additional database for managing the tenants themselves.



This mode is reserved for hosted use of the solution, managed by a partner in a SaaS fashion. A specific client application allows this partner to administer each sub-tenant in terms of available features, licensing and accounts provisioning. Once declared in the multi-tenant system, the tenants are isolated from each other via separate databases and administrative interfaces. Each tenant is logical to a single-tenant system with the only difference being that they cannot operate the license renewal themselves (entrusted to the partner who administers the hosting).

Extensibility and interfacing

Ofelia® is a scalable application which can be easily interfaced with external systems. Interfacing can be done in different ways:

- **Use of specific drivers:** the software has about 25 standard drivers that allow it to connect with many protocols and solutions on the market (see *APPENDIX I: Drivers' configuration* on page 219). New drivers are regularly developed and provided along with each software update.
- **Use of external applications:** these plug-in components are dynamically loaded into Ofelia® and can be used to address interconnection issues with third-party software. External applications can also be used as identity providers, allowing the user to authenticate in the software with external credentials (for instance, to enable *Single-Sign-On* with an Active Directory), or as import gateways (for example, to import rooms and devices for an entire facility from a TeleCARE® system).
- **Use of Service Hooks (WebHooks):** to dynamically notify a compatible system (i.e., one that already exposes a REST API) of changes in event status. This mechanism is detailed in the administrator guide (see *WebHooks* on page 86).

User Manual

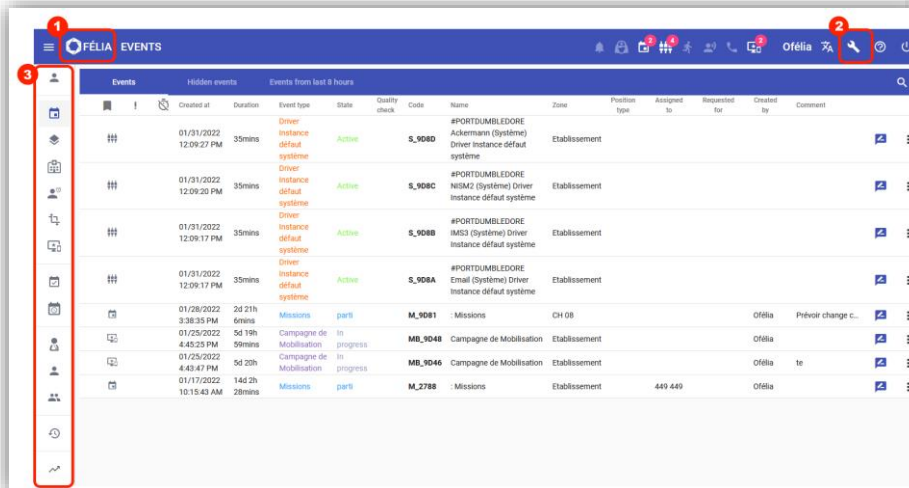
General Guidelines

Ofelia's[®] graphical interface is accessible through a web browser and requires a connection from the user. Each connection delivers an access token that is valid for 20 minutes and is renewed for each action in the software. In case of prolonged inactivity beyond this limit, the user is redirected to the login screen where he/she must authenticate again.



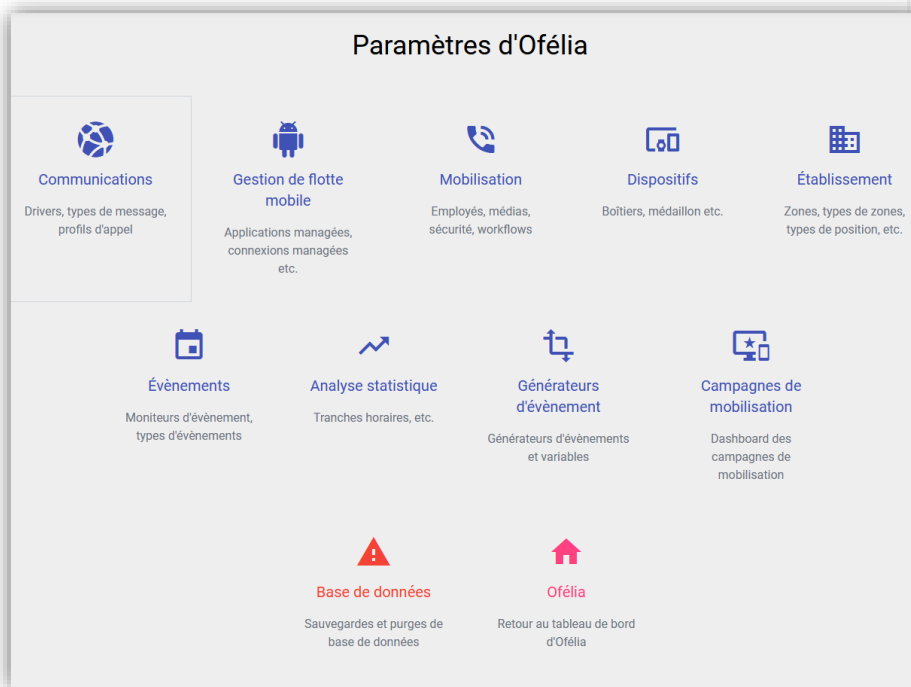
The web application itself is divided into two main areas, one dedicated to the configuration of the software and the other to its operation. The set of permissions allows to grant selective access to each area or functionality for different categories of users. In each area, the functionalities are divided by themes, each of them having a vertical menu to access directly to its own features:

- 1 Access to the Operation area from the Configuration area
- 2 Access to the Configuration area from the operation area
- 3 Vertical menu for accessing features



Administrator Guide

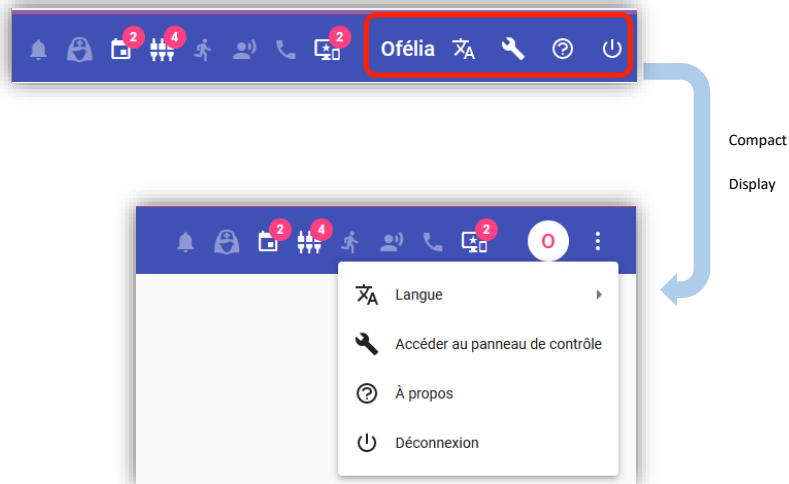
Due to the large number of configurable elements in the software, the configuration area is itself broken down into several zones, giving a simplified access to the various parameters. Upon entering the area, the administrator is presented with a welcome screen hosting several shortcuts to the different sub-areas.



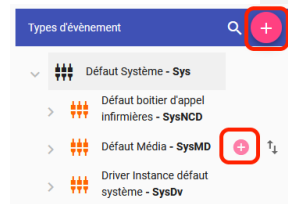
Depending on the licenses activated on the site, all the options and functionalities will not necessarily be visible, as well as each functionality may or may not be available to the current user depending on his permissions.

User interface common concepts

The application adapts to the target screen to allow an optimal display in all legibility. Thus, on smaller screen sizes, all the menus will not always appear as an exhaustive list but can be 'folded' into a more compact form. For example, on the home screen, some items can be accessed from a drop-down list if the width of the screen does not allow all the information to be displayed in a satisfactory manner:



For screens that allow to add main elements (users, drivers, media, etc.), a large plus sign (+) allows you to add a new item, while a smaller plus sign (+) allows you to add sub-items to a list.



Most destructive operations (like deleting an item) require prior confirmation. Error messages or contextual explanations are systematically displayed in a central notification area at the bottom of the screen.



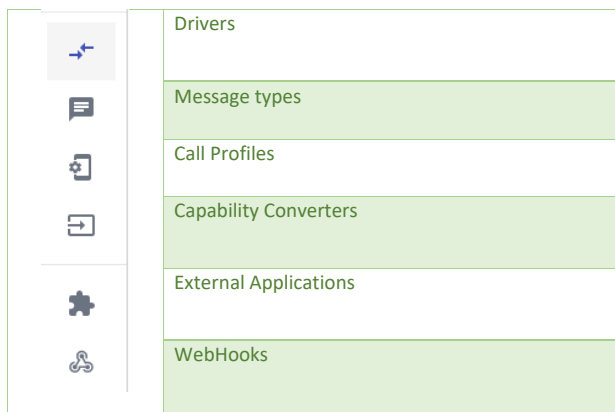
Initial setup

During the first setup of a site, it is necessary to create many elements in order to describe the structure of the facility, the equipment and peripherals deployed, the employees and their assignments, the type of events handled and the associated requisition scenarios, and so on. To carry out this configuration, the following order of parameterization is suggested in order to avoid going back and forth between the various elements and to respect the prerequisites (for example, a triggering rule cannot be defined if the types of events have not been initialized beforehand):



Communications

The communications sub-area lets the administrator define the input/output connections between Ofelia[®] and *OfeliaCom* modules: this involves indicating how to exchange with the outside of the system in terms of drivers, message types and converters, or by extending the operation of the software through service hooks or external applications. These are the lowest level settings, directly related to the peripherals and equipment connected to the software. The vertical menu gives access to the following features:



Drivers

This feature allows to define and deploy new driver instances on OfeliaCom communication modules. The administrator chooses the driver concerned (see *Drivers and driver instances* on page 22) and gives a name to the instance created. He can then enter the specific parameters of the driver for each of the deployed communication modules (by default, in a single-server architecture, the parameters are entered on the only existing module, bearing the name of the server where Ofelia[®] is located).

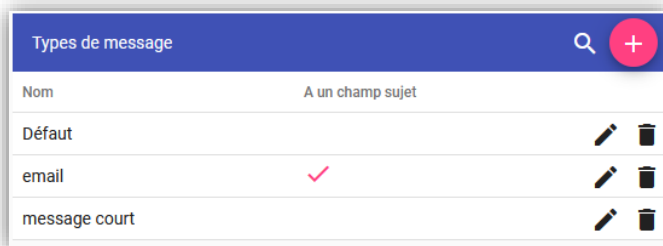
Module name

Module specific parameters

The details of the specific parameters for each driver are given in the Appendix (see *APPENDIX I: Drivers' configuration* on page 219).

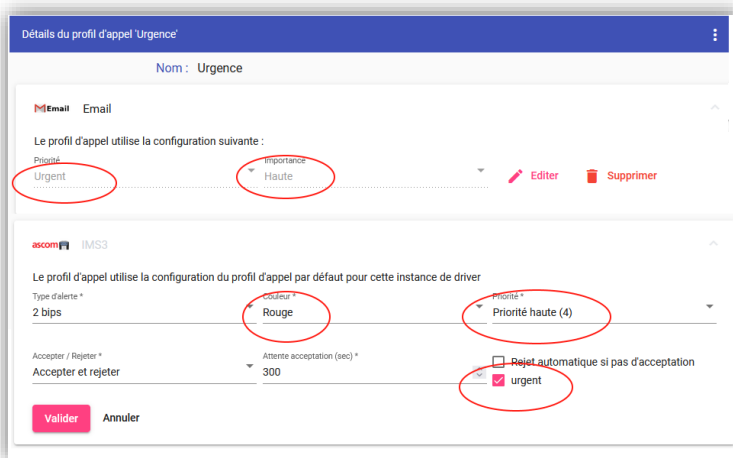
Message Types

This screen allows you to define the different types of messages that can be associated with the drivers and the media. The administrator can thus define for example a short message type for an SMS driver, and a long message type for emails. An additional attribute allows to indicate whether the message type supports the use of a Subject (for an email message for example, or as a title of the message lists on some phones).



Call Profiles

Through this configuration screen, the administrator can create a new call profile or edit an existing one (see *Call profile* on page 41) and define the characteristics that each driver instance allows to set at the profile level: for example, text color, acknowledgement timeout, etc.



The details of the properties that can be configured on a profile depend on the implementation of each driver; the list is given in the Appendix (see *APPENDIX I: Drivers' configuration* on page 219).

Capability Converters

This screen allows an administrator to create or edit capability converters which transform

- One nature of event into another (a Nurse call into a LWP alarm for instance)
- An input text to a nurse call event type,
- An input text into a variable value

Nurse Call to LWP Alarm

The administrator must map the states of the Nurse call received as input with the LWP alarm states to be converted as output.

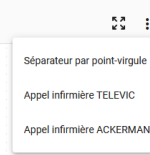
The states of the Nurse call (*Call*, *Presence*, *Assistance* and *Emergency*) do not all have to be filled-in, depending on the business context of the solution implemented.

Text to Nurse Call

The translation rule of the text frame received in a nurse call is edited via a block programming area (*Blockly*®).

In order to facilitate the writing of blocks for the most common cases, a menu located in the upper right part of the programming area allows to start from a predefined code model for frames with content separated by semicolons, as well as for the structure of ASCII frames of *TELEVIC*® and *ACKERMAN*® Nurse calls units.

These templates can be used as is or as a starting point to write your own conversion logic, without having to start from a blank page.



Text To Variable

The administrator can use the *Blockly*® programming area (see below) to define how to extract information from a text frame received as input and use it to define the value of one or more variables. In turn, modifications to these variables will allow the creation and evolution of other event states (alarms, etc.). As with the programming blocks used to transform a text into a Nurse call, the programming area allows you to create a logic from scratch by assembling blocks, or from any of the built-in models: extracting text separated by semicolons, or using the *ESPA*® or *OAP*® standard message structure.

Blockly® Programming

The *Blockly*® programming mode is an open-source solution developed by Google®, which consists of placing blocks of code on a programming area by assembling them graphically according to the logic expressed. The blocks of code belong to different categories:

Category	Content
Logic	Conditional tests, logical operators (AND, OR, etc.), value comparisons
Math	Arithmetic, numerical functions (trigonometry, square roots, modulo, etc.)
Text	String manipulation, casing, pattern extraction
Lists	Operations on arrays
Variables	Operations on Blockly® variables (not Ofelia® specific variables !)
Regex	Regular Expression manipulation (pattern recognition, extraction and substitution)
Ofelia	Operations on Ofelia® proprietary variables, changing device states

More information about *Blockly*® block programming, in a broader context than Ofelia®, can be found online: <https://developers.google.com/blockly>

Third-Party Applications

The presence of these third-party applications depends on the license granted, and each configuration is specific. It can be a matter of configuring the access accounts to the remote services as well as the data import and export mappings.

WebHooks

Service hooks are extensibility points in the software that allow, when certain conditions are met (like a call completed, change of state of an event, etc.) to initiate an http call (POST) to an external web service. The administrator can configure as many service hooks as he wants, specifying for each one:

- Under which conditions it should be triggered
- The characteristics of the web service to be called, including its address, its security features (authentication, signature, encryption) as well as the additional headers to be filled in during the call
- The failover mode to apply in case the remote service cannot be reached

Each webhook thus defined also presents a time-stamped list of transmission errors associated with the error message, allowing the incident to be investigated in partnership with the remote service manager.

The screenshot shows the 'Création d'un nouveau web hook' form. It includes fields for 'Nom' (RemoteService), 'Adresse' (https://remoteService.org/ws), 'Compte associé' (Ofélia), and a checkbox for 'Ignorer le certificat de validation'. A red dashed box highlights the 'Contenu' section, which contains a list of topics with checkboxes. Below this is the 'Sécurité' section with checkboxes for 'Signer le contenu', 'Méthode d'encryption', and 'Méthode d'authentification'. At the bottom, there are fields for 'Durée avant déclenchement du timeout' and 'Délai entre les tentatives'.

1 **Address** of the remote service (an SSL/TLS address is recommended)

2 **User account** under which the system query is to be performed: this makes it possible to limit the data exposed to the outside world to the security context of a user (for example, to send only the events of a particular department)

3 If the address is in https, this option allows not to **check the certification chain validity**: to be used *only for testing or development purposes*

4 Triggering **topics**: hierarchical list allowing to refine the triggering modes in a global or fine way (for example "At each call" vs "At each *end of call for an alarm*"). The topics concerning the variables are available once the variables are created in the system.

5 **Custom headers** (set of key/value pairs) that will be sent "as is" to the remote server in the *Http Headers* of the outbound request.

6 **Optional signature** of the content: the key provided is used to perform a SHA256 signature whose hash will be integrated into an http header of the outbound POST request.

7 **Key and encryption algorithm** (AES or 3DES) of the message content (optional).

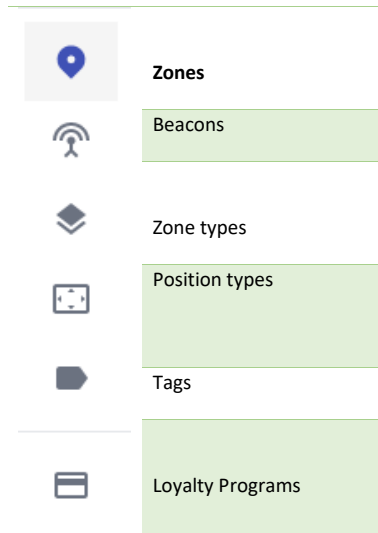
8 Optional **authentication method** to the remote service: if specified, it is either a *basic authentication* (login/password pair) or an *OAuth* token (whose validity must be checked by the remote service).

9 The **resiliency policy** allows to indicate after how many seconds a call that has not returned a success code (HTTP 200) is considered a failure. 10 The administrator can

then specify time limits for retries (e.g., after 5 seconds, then 30, then one minute) before considering the failure as final.

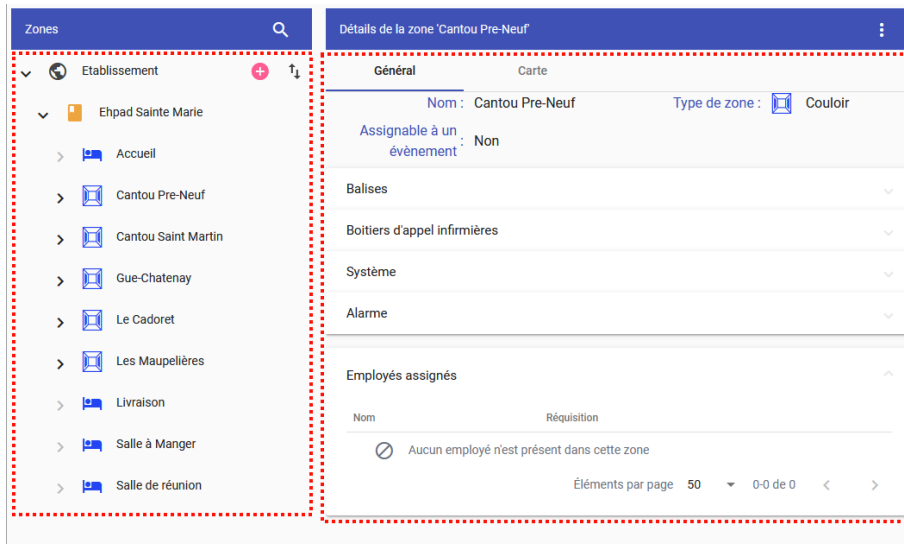
Facility

This sub-area allows the administrator to describe the structure of the facility and the beacons it contains. The features accessible through the vertical menu are the following:



Zones

The zone administration screen consists of a hierarchical tree in the left pane and a detail area on the right. The latter reflects the details of the item selected in the tree and allows to edit its contents. It has two tabs: one giving access to the main attributes of the zone as well as the list of elements attached to it (beacons, nurse call devices, alarms, etc.) and the second allowing you to locate the children of the zone graphically on a map.



The zone hierarchy tree allows to add a child at any level, move a node to another location in the tree, or delete a node. These operations are available through the appearance of icons next to the selected node: to add a child and to move the node to another location. The root of the facility must always be unique (the root node in the image shown here).

Every zone must therefore have a parent (except for the root node), a name and a zone type. It is also possible to add other parameters. Some elements of the zone screen allow to directly create the objects attached to the zone without having to go through a specific screen: for example, beacons or nurse call boxes can be directly created from the zone screen, without having to access the dedicated screen.

The hierarchy of zones in the facility allows for the inheritance of certain properties. For example, a department defined on a parent zone will automatically be applied to all child zones: thus, if the administrator indicates that a corridor belongs to the Cardiology department, then all the rooms in the corridor will also belong to this department. Also, if it is changed on the parent, the change will be immediately reflected on all descendant zones. This inheritance applies to all depth levels and can be modified for some of the children: each time a software property is inherited from one of its parents, the inheritance link can be broken thanks to the icon : in our example, the administrator can thus break the inheritance of some rooms in the corridor to assign them to the *Pediatric* department despite what is specified at the higher level. On the other hand, the icon representing a chain () allows to recreate the inheritance link and to benefit again from the attributes of the parent on a property which had been modified, thus discarding local changes.

New zone creation

The creation of a new zone is done by means of the contextual icon that appears next to the desired parent for the zone.

The editable properties are the following:

- 1 **Name of the zone:** it is mandatory and must be unique in the whole facility
- 2 **Code:** optional attribute to give a short name to the zone (can be used for example in messages to media that have little display space).
- 3 **Camera stream Url:** URL (http/s) to display the video stream of a camera located in the zone.
- 4 **Assignable to an event:** If this box is not checked, the zone will not be able to host an event. This makes it possible to have purely structural zone levels (corridors, halls, etc.) that do not impact the event management process. For example, you may want to organize the rooms of a hospital into sectors without being able to associate missions or calls directly to these sectors.
- 5 **Zone Type:** This mandatory field qualifies the zone according to a type created earlier (see *Zones, zone types and positions* on page 45). The zone type can be used elsewhere in the software (for example, it can be used as a restriction to trigger certain scenarios only if the event occurs in a room).
- 6 **Unit:** Department to which the zone belongs (aka Medical Unit, Section, etc. depending on the context). This element is inherited from the parent and the inheritance can be overloaded (7) for the zone and its children.

Once the zone is created, these parameters can be modified, and it is possible to directly add linked elements attached to this zone:

Adding a Beacon

Adding a beacon requires filling in a unique name 1 and identifier 2, as well as optional properties, such as its description 3.

4 By indicating that a beacon is **Accurate** or not, the administrator qualifies the quality of the positioning that can be obtained by this beacon: for example, a *beacon* type will be more precise than a *GPS beacon* or a positioning by *Wi-Fi*.

5 The **exclusionary** beacon specification addresses a very specific need to indicate when a loss of location will occur. Let's take the example of an interior beacon located near an exterior wall of a building: it is very likely that an employee located in the parking lot near the window of the room where the detector is located is - wrongly - considered to be present in the room. To compensate for this inaccuracy, it is possible to place an '*excluding*' beacon just after the exit door of the building's entrance. When the user crosses this threshold, he is detected by the excluding beacon and his position is no longer analyzed. Conversely, another *non-excluding* beacon, this time located inside the building's entrance, will detect the employee's return to the premises and resume his or her positioning: the excluding beacons thus make it possible to define a perimeter of validity for the other beacons. Once these beacons have been crossed, the staff members are temporarily excluded from the localization system.

Adding a nurse call device

Adding a nurse call box directly from the zone detail window is done in the same way as for adding a beacon. The elements to fill in are given below:

- 1 Unique **name** of the box (mandatory), it is recommended to establish a naming convention to easily identify the boxes later.
- 2 **Telephone number**: optional, this number is only used in the case of an addressable box with an embedded telephone grid.

- ③ **Zone:** location of the box, this element cannot be modified in this context because the window is opened from within the detail of a zone.
- ④ **Position type:** position of the device inside the zone. This precision allows the box to be positioned precisely, for example at the bedside, at the door, in the toilet, etc.
- ⑤ **Device Id:** unique identifier of the device, depending on the manufacturer and the configuration of the nurse call system. This mandatory property is essential because it makes the link between the status information received by the driver instances and the device defined in Ofelia®.
- ⑥ **Driver Instance:** name of the driver instance responsible for communicating with this device to receive changes in its state.
- ⑦ **Nurse Call Type:** type of event that will be generated when a new call is created from this box. It can be the 'Nurse call' type provided by default with Ofelia® or a child type that has been created by the administrator (see *Event Types Edition* on page 128). If the selected event type has parameters, an interface appears to enter each of them.
- ⑧ **Visit Reminder:** Name of a visit reminder policy defined by the system and to which this device must conform. This visit reminder can allow, for example, to automatically put the box on call if it is inhibited and if no presence has taken place in the room for a defined period of time (see *Visit reminders* on page 48).
- ⑨ **System Event Generation:** indicates if you want a failure of the box to automatically create a system event (which could in turn become a source of new scenarios).
- ⑩ **Controllable device:** if the device is controllable, it is necessary to indicate an address ⑪ as well as the driver instance name ⑫ in charge of piloting this device and controlling its call and presence states.

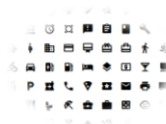
Adding an alarm generator

The addition of an alarm generator in a zone is also accessible directly in the zone detail window.



The parameters to be filled in are the following:

- ① Unique **Name** of the generator, allowing to identify it later
- ② **Icon** to visually identify the generator, to be selected from a predefined list
- ③ **Zone:** from the detail screen of a zone this field cannot be edited, it corresponds to the zone currently selected
- ④ **Event Type** to generate: it can be the system event type provided by default with the software or a child type created by the administrator. If the selected event type has parameters, an interface appears to enter each of them.

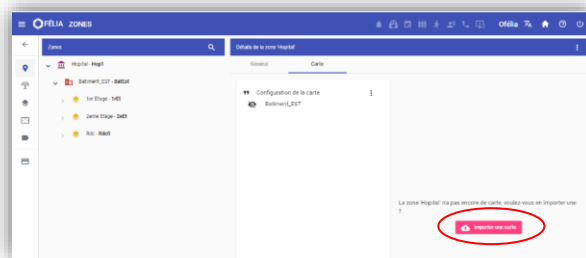


Floor maps managements

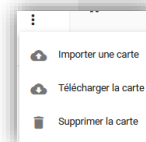
A dedicated tab in the zone administration screen allows to assign to each level of the tree structure a graphic representation in the form of a vector map. The administrator can then associate each child zone and each tag to an area of the map so that the software can color it when an event occurs. This configuration is a prerequisite for the use of the Floor map events view (see *Floor map view* on page 187).

To ensure proper configuration of the plans, it is imperative to:

- Use only *.svg* format drawings (vector drawings). These must respect the prerequisites defined in the paragraph *Vector Files Format* below.
- Fill in **each and every** level of the tree structure, from top to bottom, for example: site, building, floor. The absence of a graphical representation at a level of the tree structure will prevent proper navigation from the interface.



If no map is defined on an area that has children (zones or tags), the "Import a map" button allows to upload a new graphic file. If a map already exists, a contextual menu (vertical dots ) allows to delete it, to download it to the client computer (for archiving purposes for example), or to replace it with a new map.



Vector Files Format

SVG files are a structured representation of a vector drawing in the form of geometric structures (paths, polygons, etc.) expressed in a tag language (XML). They can be created with any compatible tool on the market (*Adobe Illustrator*® for example), provided that a few rules are respected, allowing Ofelia® to identify the useful items of the drawing.

The tags representing details on the map and having no specific functionality are not subject to any particular rule. On the other hand, the main tag of the file, as well as those that will be assigned to elements of the software (floors, rooms, tags, etc.) must comply with a few structuring rules:

SVG Tag

This tag located **at the very beginning of the file** must be unique and have the following 3 attributes:

- `class="map"`: let Ofelia® understand that the SVG file is a map

- `viewBox="0 0 1200 800"`: defines the size of the map, it is forbidden to specify an attribute X or Y, *Width* or *Height*, in pixels. However, the value remains free depending on the desired rendering of the map.
- `width="100%", height="100%"`: forces the map to occupy all the necessary space

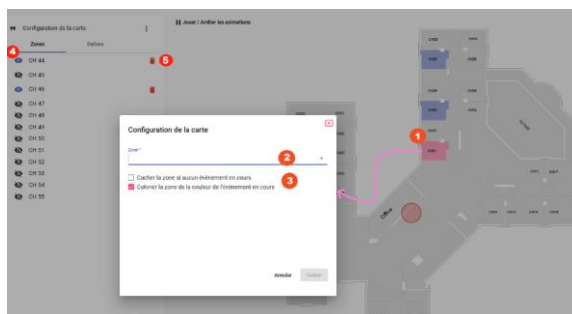
Functional Tags

For a tag (rectangle, polygon, path, etc.) to be associated with an element declared in Ofelia® (a corridor, a room, etc.) it is imperative that it sets the following attributes:

- `class="zone"`: allows Ofelia® to recognize the assignable areas on the map
- `data-map-id="xxx"`: allows Ofelia® to distinguish the different zones within the same map. The **xxx** identifier must be unique for each zone on the map.

Definition of zones on the map

Mapping areas on the map consists of selecting on the drawing the polygon representing the area to be associated, then choosing from the list of children of the area the one that corresponds to this location. Each selectable area is highlighted when the mouse pointer passes over it.



- 1 By clicking on a highlighted area, the window for associating it with a zone pops up.
- 2 Choice of the zone to be associated with the selected polygon, among the direct child zones of the zone being edited (represented by the map)
- 3 Options of the area on the screen: it is possible to
 - Hide the entire polygon when no event is taking place (e.g., for a tag on which you have placed a polygon containing an animation that you only want to see when it is activated)
 - Color the area on the plans with the color of the type of event taking place there: in this case the area will appear unchanged as long as no event is taking place.
- 4 When an area is defined, clicking on the eye icon  in the list will highlight it to locate it on the map.
- 5 The  icon removes an association already made between a zone and an area on the map

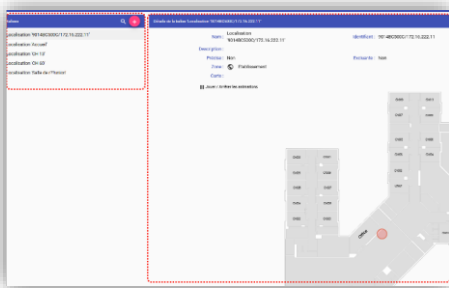
The administrator can proceed in the same way to place the localization beacons, in the tab provided for this purpose.

Localization Beacons

The beacons administration screen allows you to edit the characteristics of each of the beacons present in the facility, and to position them on the map of their parent zone.

Selecting a beacon from the list on the left of the window displays the details of the tag and its characteristics directly opposite in the detail area on the right. It is also possible to add a new beacon using the button  at the top of the list.

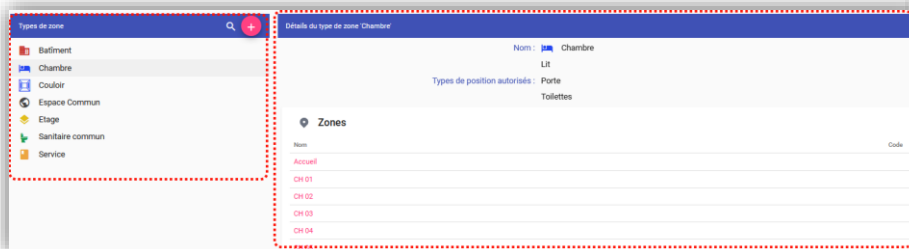
The contextual menu at the top of the detail page (vertical dots ) allows the beacon to be deleted or its attributes to be edited. The editable properties are those described in the chapter on adding a beacon to a zone, with the exception of the zone, which can be modified here (see *Adding a Beacon* on page 90).



The SVG file used as a map for the zone where the beacons are located is displayed if it has been previously entered in the zone administration. On this file, some graphic elements may contain animations (for example, the coverage area of a flashing beacon). An option allows you to pause these animations to improve readability.

Zone Types

The zone type setting screen is divided into a selection list on the left, containing the types already defined in the system, and a detail area displaying information about the zone type selected in the list. A pop-up menu (vertical dots ) allows you to delete or edit the selected zone type, while a button  at the top of the list allows you to create a new zone type.



When the detail area is in view mode, the list of areas belonging to this type is displayed for information purposes. When creating or editing a zone type, the following properties can be set:

The screenshot shows a form for creating or editing a zone type. It has five numbered callouts: 1 points to the 'Icône' field, 2 points to the 'Nom *' field, 3 points to the 'Couleur' field, 4 points to the 'Types de position autorisés' field, and 5 points to the 'Tags' field.

- 1 **Icon** to be selected from a predefined list
- 2 **Name** (mandatory) to uniquely identify the type of zone
- 3 **Color** used to display the zone type
- 4 **Allowed position types**: selection of one or more position types from the list of those defined (see *Position Types* below), for example 'Bed' and 'Door'. These are position types that can be assigned to users (patient, client) or to devices (e.g., nurse call device).
- 5 **Tags**: A list of labels associated with the zone, among those defined globally in the system. For example, an administrator can add a "COVID Zone" label to certain rooms to adapt requisition scenarios with additional constraints.

Position Types

The list of position types is a list that allows you to define, within a zone, all the positions that can be occupied by patients (or clients) or devices. This list is hierarchical and allows, for example, to define a position type 'Bed' that is the parent of position types 'Window Bed' or 'Door Bed'. The various rules, filters and restrictions applied to the 'Bed' position types will then also apply to each of the child types.



The position type administration screen shows a tree list on the left and a detail area on the right for the currently selected element. A contextual menu (vertical dots ) allows to delete an item or to edit its content. Finally, a button allows you to add a type of position at any level of the tree structure or to move a type under another parent. It should be noted that it is possible to have several root elements in this tree structure.

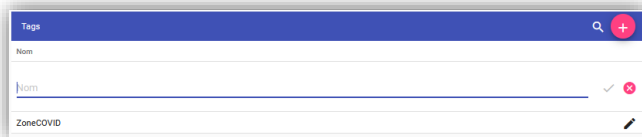
The list of editable properties on a position type is limited to its **name**, which is mandatory and must be unique in the software, and its **capacity**. If the position type is a bed, the capacity will usually be reduced to 1 in a hospital context, but could be 2 in a hotel organization. Similarly, a zone type *'Toilet'* will have a capacity of 0 (zero) because the capacity is only significant in terms of stay and accommodation.

This capacity can also be used to calculate the number of patients (resp. clients) that a zone can accommodate, depending on the position types represented in the area: for example, a double hospital room with the position types *'Bed Door'* (capacity 1), *'Bed Window'* (capacity 1), *'Door'* (capacity 0) and *'Toilet'* (capacity 0) can accommodate a total of 2 patients. In this case the position types *'Door'* and *'Toilet'* will only be used to identify call devices.

Tags

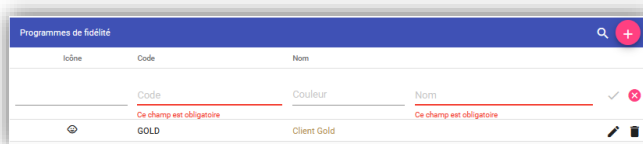
This very simple administration screen allows you to add, edit and delete labels which can then be added to other elements of the software (in particular the zone types, see on page 95).

The list, made up only of keywords which must be unique, can be administered very simply by means of contextual buttons.



Loyalty Programs

Loyalty programs are a specific feature for managing hotel customers. Compatible with the Opera® import gateway (see *Opera Driver* on page 233), it allows you to associate a loyalty level (*Gold*, *Premium*, etc.) with customers defined in the system. As with the tags, all the administration (deletion, creation, editing) can be done directly via contextual buttons in the list of programs. The information to be filled in is minimal: a unique code and name, as well as an icon and a color.



Mobilization

This sub-area contains all the elements required to program the requisition cycles. The vertical menu gives access to the following functions:

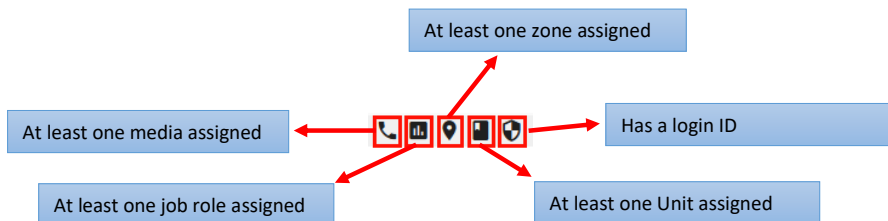
	Staff Members
	Plannings
	Job Roles
	Services
	Presence Types
	Availability Levels
	Availability Profiles
	User Profiles
	Files
	Sound Profiles
	Media Templates
	Media Models
	Medias
	Roles and Permissions
	Workflows
	Triggering Rules
	Call Strategies

Employees

The employees are a central element of the requisitioning system: it is indeed through the knowledge of their assignments, presence and location that the various scenarios will be able to call the most appropriate people in a given situation.

The configuration of employees therefore includes numerous properties, which can be managed through a single configuration screen. This screen is presented in the form of a list of employees on the left, with indicator icons (detailed below) for each one, and a detail area that appears on the right next to each selected employee.

The list shows the following icons next to each employee's name and surname, giving an instant status of their configuration:



To add a new staff member, the following properties must be filled in after pressing the add button **+** :

The screenshot shows the 'Ajout d'un nouvel employé' form. On the left is a list of employees with a search bar and a '+' button. The form fields are numbered 1 through 11:

- 1: Nom *
- 2: Prénom *
- 3: Identifiant de carte
- 4: Langue préférée
- 5: ☐ Activer la connexion permanente
- 6: Type de présence
- 7: Profil utilisateur
- 8: Alarme DATI (Désactivé)
- 9: Type d'alarme DATI
- 10: Template de média
- 11: Profil de disponibilité

Buttons 'Valider' and 'Annuler' are at the bottom.

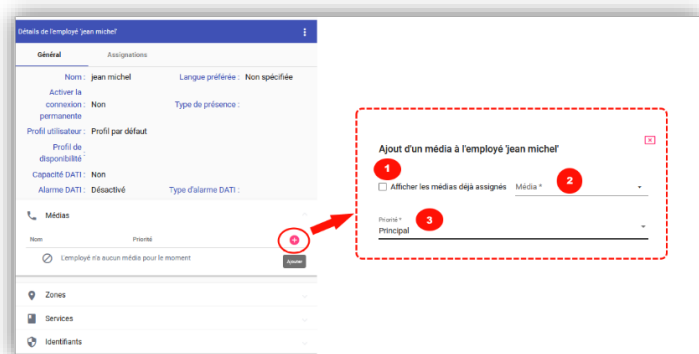
- 1 **Name:** employee's last name or customary name. Mandatory.
- 2 **First Name:** first name (mandatory for display purposes)
- 3 **Card Id:** allows you to associate a staff member with an identification card holder. In the case of presences realized by badge on nurse call box (ASCOM®), the employee will then be automatically associated to the event state transition.

- 4 **Preferred Language:** For information purpose only, can be leveraged to customize the language used in interactive messages or in graphical interfaces.
- 5 **Enable permanent connection:** If this box is checked, the web interface keeps the user's connection open.
- 6 **Presence Type:** Optional, allows you to identify the user who is performing a presence with his badge by giving him one of the presence types defined in the software (for example 'Doctor' or 'Nurse'). For additional details, please refer to *Presence Type* on page 36.
- 7 **User Profile:** Allows the administrator to assign the employee to a user profile that governs display preferences (page size, landing page on login, etc.). For more information on user profiles, see on page 113.
- 8 **LWP Alarm:** determines the behavior of the lone worker protection alarms for the employee. The options are: **Disabled** (no alarms supported regardless of media used), **Follow media capabilities** (alarms supported only if the user has a media that supports LWP alarms. This option, which allows a single LWP license to be shared among several users, may not work if the number of users connected exceeds the number of licenses assigned), **Forced** (maintains the LWP function active, this consumes a LWP license in all situations).
- 9 **LWP Alarm Type:** Type of event to be generated when the employee triggers a lone worker protection alarm.
- 10 **Media Template:** If the user has a media compatible with Templates (for example an *OfeliaSmart* media), the selected template allows to define a default behavior of the device.
- 11 **Availability Profile:** Profile to be used to interpret how the user should react to each event priority according to his availability (see *Availability Profiles* on page 111).

Once the employee is created, it is possible to associate a number of related items directly from the employee details window: assign media to the employee, assign the employee to zones or departments, or define the employee's login credentials to the application. In the same way, the editing window of an employee gives access to the definition of his schedule in a separate tab.

Assigning a media to the user

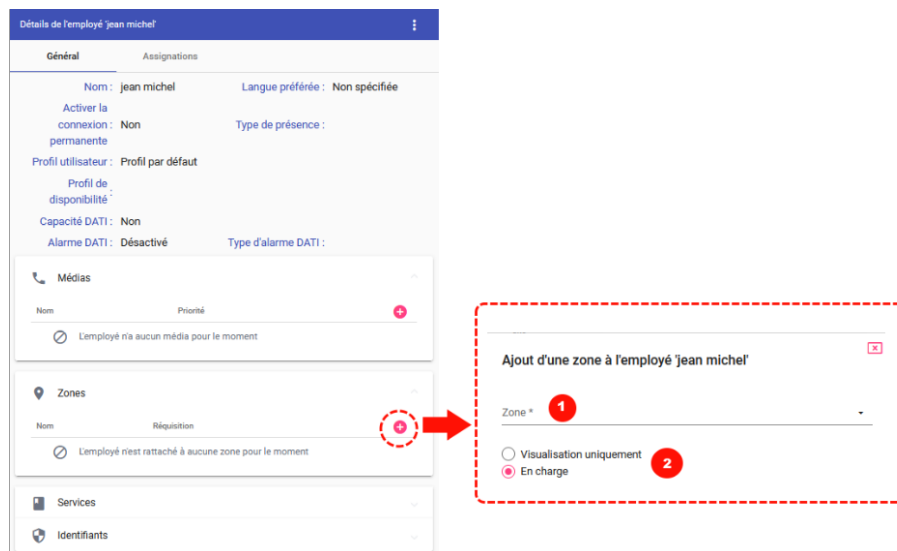
From the employee's details, the administrator can delete or add media assignments. To do this, a contextual add button (+) opens a selection window among the media defined in the system (see *Medias* on page 118).



The media can then be chosen from a list ², in which by default only the media that are not already assigned to a user are proposed (an option ¹ allows this behavior to be changed). Finally, a priority must be defined for this media among all those assigned to the user: ³ main media or media used in 1st fallback, second fallback, etc. This precision will allow you to define the order in which messages are sent if the employee can be reached on several devices.

Assigning a zone to the user

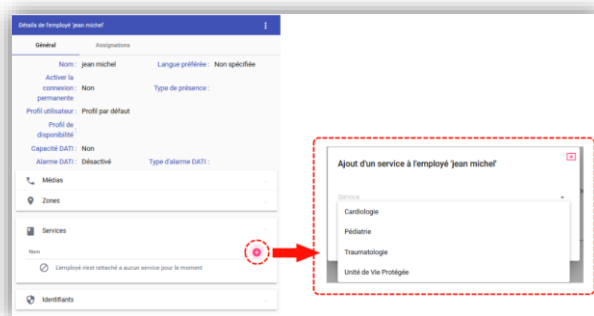
From within the employee's details, the list of zones the employee is assigned to is displayed. From this list, it is possible to add, edit or delete assignment zones.



Adding or editing a zone consists in selecting the concerned zone in a drop-down list ¹ and indicating if the employee ² has only the right to view the zone (without being responsible for it, only for visibility or supervision purposes) or if he/she is *in charge* of it (in which case he/she can be requisitioned automatically if an event occurs in this zone or in one of the child zones).

Assigning a Unit to the user

From within the employee's details again, the list of departments (aka medical units, or sections) to which the employee is assigned is displayed. From this list, it is possible to add, edit or delete unit assignments, exactly as it is done for the zones. However, a unit assignment is even simpler because it is only a matter of associating the employee with the department, without any notion of hierarchy or personnel in charge or in simple visualization.



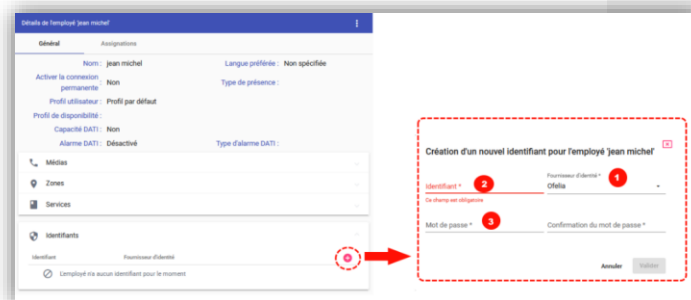
The selection is made directly in a list of departments, sorted alphabetically by their name.

Assigning a Login ID to the user

Also from the employee's detail window, the administrator can administer login credentials, revoke them, change passwords, etc.

Ofelia® possesses its own identity provider (as described on page 39) and can also be enhanced with additional 3rd party identity providers (*Windows Active Directory*® or LDAP for example). Depending on the identity provider chosen, requested information may vary.

In the case of the integrated provider, adding an identifier consists in choosing a unique login name ² as well as an associated password ³.



Schedules

The management of schedules is carried out in two stages: firstly, the definition of schedules used by all teams (for example, the schedule for public holidays, the definition of a three-shift or night/day work cycle, etc.) and then the assignment of employee functions to these schedules.

The objective of the schedules is to allow the system to determine for each employee, when an event is triggered, whether he or she is:

- ⇒ Absent
- ⇒ Present for one or more specific functions (e.g.: Doctor, Fireman, etc.)

The purpose is to have a system that indicates which function(s) each user occupies at a given time.

Schedule Creation

Even before defining the time slots, the administrator must create the general definition of the schedule. This involves defining the purpose of the schedule (absences or on-call duty) and its periodicity by modeling the organization of work in the establishment.

A schedule is characterized by a name (mandatory and unique) as well as a period of applicability and a type. This schedule type reflects its periodicity and can take the following values:

- **Daily:** Allows to define a typical work day. For example, from 8am to 11am in the morning and then from 2pm to 5pm.
- **Weekly:** Allows to define a typical work week. For example, Monday to Friday from 8am to 11am then from 2pm to 5pm, and nothing on Saturday and Sunday.
- **Fortnightly:** Allows to define a fortnight, for example if, as in some facilities, the organization of work differs between 'even' and 'odd' weeks.
- **Monthly:** defines a monthly work organization, more seldom this way of defining schedules is reserved for particular organizations based for example on long cycles (3 weeks on site and 1 week of off).

Whatever the type of schedule, it is intended to be repeated according to a predefined periodicity: daily schedules are repeated every day between the application start date and the end date, weekly schedules every week, etc. Though the start date of applicability is mandatory (by default the current date), the end date of application may be left unspecified: in this case the schedule will be valid from the start date and without limit of duration.

A button at the top of each column (2) lets the administrator add a new time slot on the corresponding day:

The start and end times are mandatory. If the minutes are not filled in, they will be considered as 0 (zero) and the time slot will fill a full time slot.

- 2 **Time Slot Type:** one of the types previously defined by the administrator.
- 3 **Slot meaning** (see *Absences and on-call duty* on page 37):
 - o **Standard** (default): the range is assigned to the user in the slot. If the user has several assignments on the same slot, these are cumulated.
 - o **Absence Schedule:** the slot is excluded for the type of slot on the period. If the employee is otherwise present for the same function on another schedule, this absence cancels his assignment. This type of slot is useful, for example, for vacations, which thus have priority over standard assignments.
 - o **On-call Schedule:** the slot is forced for this type of time slot, with an even higher priority than the absences. If the user is on call for a function during the period, he can be requisitioned even if he has an absence period for the same function at the same time.

Assigning a Job to the user

Once the schedules have been defined, the administrator has a broad representation of the different work organization modes in the facility (standard schedules, holidays, night shifts, etc.). The only thing left to do is to assign a function to the employees on these different schedules, and possibly adjust them on a case-by-case basis to reflect the actual availability of each person (for example by adding absences for vacations).

These assignments are made from within the detail window of each employee, in the "Assignments" tab.

This view has two main areas: the top area (A) is for managing full-time assignments, while the bottom area (B) supports more specific assignments on a calendar.



Full-time assignments are a simple flat list of functions that the employee occupies for the entire time of his presence in the facility. For a simple management of requisitions, it is possible to use only this mode of assignment: the administrator assigns to each employee one or more functions (doctor, nurse, guard, etc.) and the requisition engine will use this list independently of the event schedule, simply based on the availability of the user and his media. A pop-up icon next to each line allows to delete the corresponding assignment.

Calendar assignments are made on the calendar view displayed at the bottom of the screen, whose general characteristics are as follows:

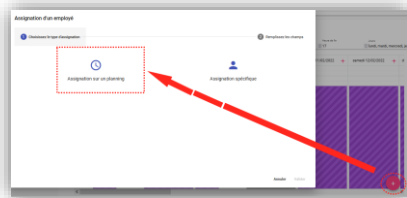
- 1 Navigation of the date range displayed on the screen: selection of the date range, and navigation in the calendar with the arrows to move forward or backward by the length of the selected period. Allows you to navigate, for example, from week to week on the screen.
- 2 Display restrictions: choice of the daily schedule and the weekdays displayed (for example, if the facility is closed on weekends, there is no need to display a full week).
- 3 The area for each day shows the time slots occupied by each function with the color code provided for the type of slot. A double-click on a period allows you to edit its characteristics.

- 4 Absences and on-call time appear separately on the display, in the form of layers that can be overlaid on the calendar, thus making it possible to focus on the problems of managing daily life or vacations jointly or separately.
- 5 Assignment of the employee on a schedule: see below.
- 6 Specific assignment: shortcut button allowing to define a one-time assignment (not recurrent) on a time slot of the day.

Using the add button at the bottom right of the calendar (+), a choice screen allows you to select either the assignment on a schedule or a specific assignment. This second option is similar to using the add button at the top of each day, except that it does not pre-select the day of application.

Assigning the user to a schedule

This is the easiest way to assign an employee to one or more time slots defined on a schedule.



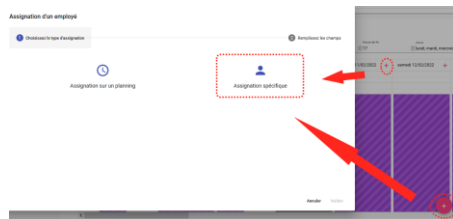
Once the option is chosen, a definition window allows you to precisely define the properties of the assignment:

- 1 **Schedule:** selection from schedules created by the administrator (see *Creation of a schedule* on page 36)
- 2 **Time Slot Type:** this is the category of time slots defined in the selected schedule. For example, it can be the 'Morning' or 'Afternoon' slot of the selected schedule (see *Creation of a schedule* on page 36)
- 3 **Function:** selection of the function/job occupied by the employee on this type of schedule slot (ex: Doctor, Midwife, etc.)
- 4 **Period of validity of the assignment:** only the start date, by default set to the current date, is mandatory. If no end date is specified, the assignment will last as long as the schedule on which it is associated.

Once the assignment has been validated, the corresponding time slots appear in color on the calendar. A double-click on any of them (regardless of the occurrence) opens the assignment definition window again to browse or edit its properties.

Custom Assignment

This type of assignment is accessible by choosing 'Custom Assignment' in the window that opens after using the add button at the bottom right of the calendar (+), or directly from the contextual add button at the top of each day.



If the creation of the assignment has been initiated from the contextual button at the top of a day, only a one-time assignment, i.e., not periodic, is possible. The only information to be filled in is the following:

Assignment d'un employé

Choisissez le type d'assignation

Remplissez les champs

1 Fonction *

2 ☐ Créneau horaire d'absence
☐ Créneau horaire d'astreinte

3 Date de début *
9/2/2022

4 ☐ Toute la journée

5 Heure de début *

Heure de fin *

- 1 **Function/Job** of the staff on the period
- 2 **Absence or on-call time slot**: by default, new assignments are considered a presence on the function. These options allow you to alter this behavior.
- 3 **Start date**: Set to the date of the add button that was used
- 4 **All day**: allows you to pre-populate schedules for a full day
- 5 **Range**: start and end times are mandatory.

To allow a more precise definition of the specific assignments to an employee, and in particular to have access to the recurrence details, it is necessary to use the add button located at the bottom right of the calendar and then to choose the corresponding option (*Custom Assignment*).

Assignment d'un employé

Choisissez le type d'assignation

Remplissez les champs

1 Fonction *

2 ☐ Créneau horaire d'absence
☐ Créneau horaire d'astreinte

3 Date de début *
9/2/2022

4 ☐ Toute la journée

5 Heure de début *

Heure de fin *

TOUTES LES SEMAINES

Toutes les 1 semaines

Lundi Mardi Mercredi Jeudi Vendredi Samedi Dimanche

Toujours

The left side of the screen has the same features as described above for a single specific assignment, but there is now a new area on the right side of the window that allows you to define a periodicity for the assignment you are creating. This allows for example to add an absence *every Wednesday afternoon every other week*, or an assignment as a first aid worker *on the first Wednesday of each month for 6 months*. The periodicities available are as follows:

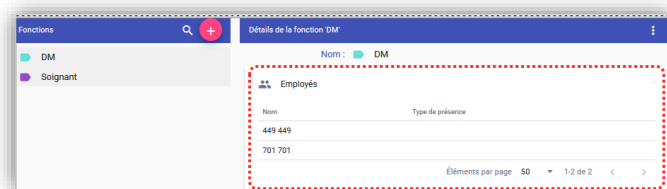
- **Every day**
- **Every week:** on specific weekdays
- **Every month:** on a fixed calendar date (e.g., the 6th of the month) or on a day of the month (e.g., the 2nd or last Wednesday of the month)
- **Every year:** on a fixed calendar date (e.g., December 25) or by day occurrence (e.g., the last Tuesday in February)

All these periodicity patterns can have an interval (for example every 4 days or every 3 weeks) and an optional end date.

Functions (Jobs)

The list of functions that can be occupied by employees is a prerequisite for the creation of staff assignments on schedules. The functions can be used in requisition scenarios (e.g., to call all present staff members occupying the function 'Nurse' when a nurse call is triggered). An employee can be assigned to several functions at the same time or on separate schedules.

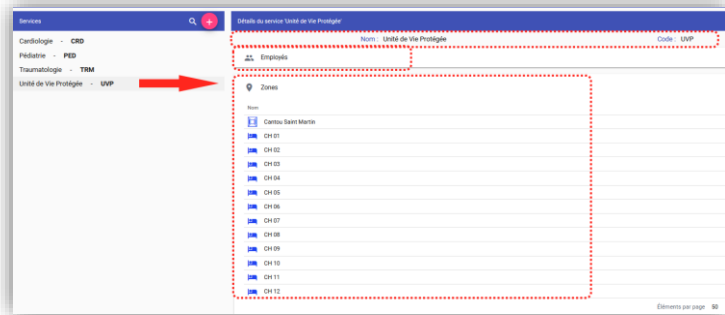
The administrator edits the list of functions by assigning them a name and a display color. Next to each function, the detail area displays the list of employees who are assigned at least once to this job (either full time or on a schedule).



Departments/Units

Departments represent medical units (in hospitals) or sections (in the hotel industry), generally speaking they are a transverse, non-hierarchical administrative qualification, to which both zones and staff members can be attached.

The left side of the screen shows the list of departments defined in the program: selecting one of them displays its details in the area on the right. This area shows both the properties of the department (name, code, etc.) and the list of employees and zones that are part of it.



An add button  allows you to create a new department, while a contextual menu (vertical dots ) allows you to edit or delete an existing one. The creation or modification of a department requires the following information:

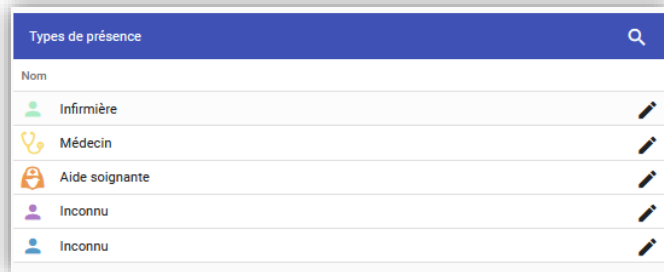
- 1 **Unique Name** of the department (mandatory)
- 2 **Unique Code** of the department (optional), used for displays and some message formats
- 3 **Visit Reminder**: Application of a visit reminder policy on all zones of the department (optional, see *Visit reminders* on page 48).

Presence Types

Presence types are a property associated with staff members which allow them to identify how they are represented when they perform a presence by means of a badge.

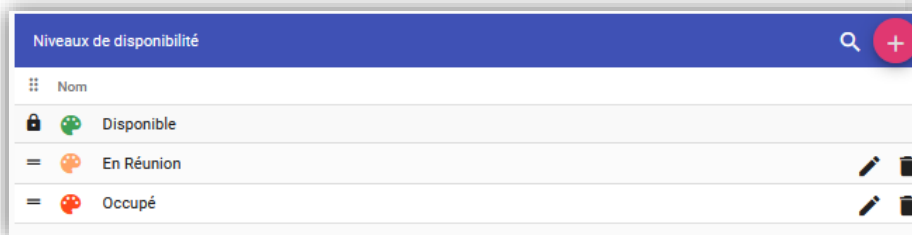
For example, on an ASCOM® nurse call device, the nursing staff can indicate their presence in a room by placing their badge on the box: Depending on the type of presence associated with the staff, their attendance in the room will be represented by an icon and a representative color in the event timeline, allowing to simply identify if it is either a nurse or a doctor.

The number of presence types is limited to 5 and is initialized with built-in default values (*Doctor*, *Nurse*, *Caregiver*, etc.). It is not possible to add or remove these built-in values, but the administrator can rename them or change their icon and color by using the editing tool  located at the end of each line.



Availability Levels

The user of the software can define at any time its level of availability among those made available by the administrator. The configuration consists in creating availability levels by simply giving them a name and a color, then ordering them by ascending order of availability (from *Available* on top, to the busiest at the bottom).



The unmodifiable level *Available* is provided and cannot be modified nor deleted: it corresponds to the maximum availability level and is assigned by default to all users when they are created. Adding a new level is done by the add button located at the top of the list, while deleting and editing are done by the contextual icons located next to each level (respectively and).

The items in the list can be reordered by drag and drop, except for the *Available* level which always remains in the very first position.

Availability Profiles

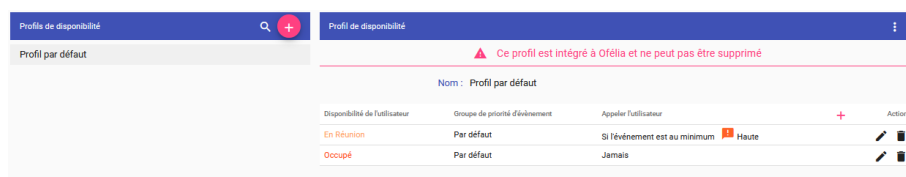
Each event in Ofelia can be qualified by a priority whose value depends on the type of event (see *Event priorities* on page 140). For example, missions can use '*Urgent*' or '*Low*' priorities, whereas alarms can use '*Imminent risk*' or '*Danger*' priorities.

On the other hand, users define their own level of immediate availability, to indicate for example that they are available, in a meeting or unreachable.

The purpose of the availability profiles is to allow the administrator to indicate how each level of availability of an employee should be interpreted according to the priorities of the events occurring in the system. For

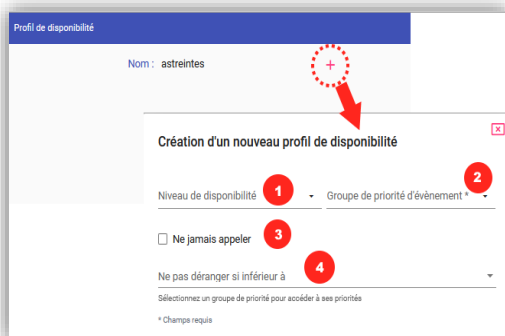
example, he can define that an employee who has the status '*In a Meeting*' should not be called during a requisition for a mission of '*Standard*' priority but that it is necessary to reach him if an alarm of priority '*Absolute Emergency*' is triggered. Moreover, each employee may require different rules: the executives will not be called for an alarm below the '*Emergency*' priority, while a reception employee will have to be requisitioned even for a '*Low*' priority mission.

To manage this plethora of customization levels, the administrator can create availability profiles that will be assigned to each employee and that will govern how the system should behave for each event priority. A default profile, which can be modified but not deleted, is provided with the software.



A list on the left side of the screen allows you to select an existing profile and to display its content in a detail area on the right side. Adding a new profile is done by the add button  located at the top of the list, deleting or editing the name of a profile is accessible through the contextual menu in the vertical zone (vertical dots ).

The creation of a new availability profile requires a name to identify it in a unique way in the software. Then, the successive use of the contextual add button allows to create each of the priority management rules in this profile:



❶ **Availability Levels:** selected from the list of those created in the software (see on page 111 above). This is the employee's availability status when an event occurs.

❷ **Event Priority Group:** selected among those created by the administrator (see *Event priorities* on page 140). These are the groups of priority levels which can be assigned to event types.

❸ **Never Call:** if this box is checked, when an event of the specified priority group occurs, users of this profile who are in this availability level will never be disturbed.

❹ If the previous box is not checked, when an event of the defined priority group occurs, the users of this profile who are in this level of availability will not be disturbed if the priority of the event is **lower than the level indicated in this list**.

Each condition defined in this way can be deleted or edited using the contextual icons at the end of the line ( and ). In the following example, a user of this availability profile will never be called during a requisition

scenario if he is in the '*Busy*' state and will only be called when he is in the '*In a meeting*' state if the event that has just occurred has at least a '*High*' priority:

Disponibilité de l'utilisateur	Groupe de priorité d'évènement	Appeler l'utilisateur	+	Actions
En Réunion	Par défaut	Si l'évènement est au minimum	Haute	 
Occupé	Par défaut	Jamais		 

User Profiles

The administrator can define the default operation of Ofelia's[®] graphical interface by assigning profiles to each user. The administration screen shows on its left-hand side a list of existing profiles, and the selection of one of them causes the display of the different characteristics of this profile in a detail area located on the right of the screen. The detail area also displays, for information purposes, the list of employees currently associated with this profile (the association itself is only performed from within each individual user edition screen).

A default built-in user profile is provided with the application. This profile can be modified but cannot be deleted. The administrator can use the add button at the top of the list to create new profiles.

The attributes that can be configured in a user profile are the following:

1 Nom ?

2 Nombre maximum de missions simultanées

3 Page d'accueil

4 Garder le menu de navigation ouvert

5 Utiliser les valeurs par défaut

6 Nombre d'éléments par page, par défaut

7 Pagination des éléments

8 Préconfiguration du filtre de la vue par services

9 Préférences de téléchargement

10 État initial

11 Activer les notifications sonores

Colonnes disponibles

Colonnes choisies

Annuler Valider

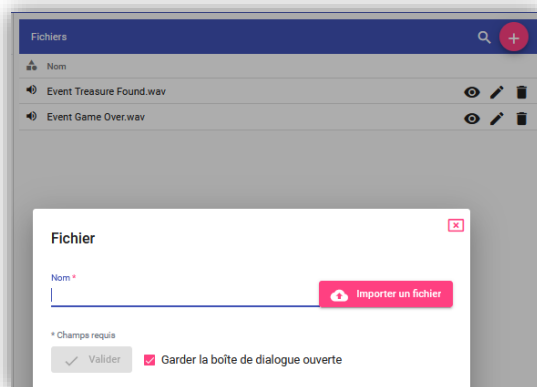
- 1 **Unique Name** of the profile
- 2 **Number of simultaneous missions**: Assigns a value that prevents an employee from being requisitioned for a mission if he or she has already reached a maximum number of assigned missions in progress
- 3 **Landing Page**: Selection of the first page that appears just after the user logs in to the application. This is one of the pages accessible in the vertical menu of the software's operating area (see User interface common concepts on page 80)
- 4 **Keep navigation menu open**: indicates whether the vertical menu is unfolded and occupies the leftmost part of the screen or whether it is folded into a version that displays only the icons.
- 5 **Use Defaults**: if this option is enabled the menu state (collapsed or expanded) is based on the default behavior of the current version of the program.
- 6 **Default number of items per page**: this is the number of items displayed by default in each page when a window containing many items is opened.

- 7 **Pagination:** list of ascending values that will appear as an option at the bottom of the dashboards to propose a different number of items displayed on each page.
- 8 **Department View:** these options govern the default operation of the event view filter by department (see Department / Unit View on page 189).
- 9 **Download Preferences:** for each of the exports planned in the program, a screen allows the user to choose the columns that will be exported and in which order, by dragging and dropping them into a list of columns (10).
- 11 **Sound Profiles:** allows the user to specify whether he wants the sounds associated with the types of events to be played in the interface, and if so, the maximum number of sounds that will be played in case many events have stacked before the program was launched.

Files

The management of binary files is centralized in the software. A single administration screen is used to load files that can be later associated with event types through sound profiles (see below). For example, the files containing the sounds that can be played automatically when an alarm is triggered.

The administration screen shows a list of files already uploaded to Ofelia® with the possibility to edit their name, delete them or preview them. Depending on the file extension that has been recognized, the preview will let the user preview an image or listen to a sound file. An add button allows to upload a new file by selecting it from the local client browser and associating a name to it.



Sound Profiles

The management of sound profiles allows the administrator to define for each type of event a sound that will be played on the user's workstation when a particular state is reached. In addition to the Ofelia® web application, other client applications can support this feature and offer contextual sound warnings, such as OfeliaSmart® or Ofelia Messenger®.

The administration screen displays on its left-hand side a list of the profiles already created, and the selection of one of these profiles displays in a detail area on the right all its properties. An add button (+) at the top of the list allows to add a new profile, while deleting or editing a profile is accessible via the

contextual menu in the details area (vertical dots ). The creation or edition of a profile consists of the administrator indicating for each state of the chosen nature the sound file that should be played and how many times the playback should take place:

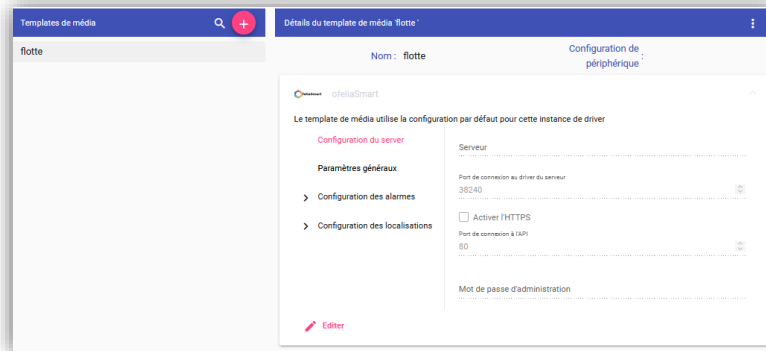
- 1 **Unique Name** of the sound profile (mandatory because it must be associated with a user profile later on)
- 2 **Nature of the event** (required because it conditions the list of proposed states)
- 3 **State** (required): state of the event triggering the playback of the sound
- 4 **File** (mandatory): sound profile previously uploaded in the 'Files' section (see *Files* on page 115)
- 5 **Number of repetitions** (mandatory, by default sounds are played only once)

It is not necessary to define a sound for each state, only the relevant ones must be indicated. This button allows to add other states possibly associated with other sounds or a different number of repetitions.

Media Templates

Media templates are a simple way for an administrator to force a configuration on a bunch of compatible devices. This notion mainly concerns the *OfeliaSmart* Android® software: the creation of a template allows to use an instance of an *OfeliaSmart*® compatible driver to force the configuration of some users' mobile devices by indicating, for example, the connection parameters, the localization priorities of the different types of beacons, etc.

The administration screen displays a list of existing templates on the left. Selecting one of these templates displays in a detail area on the right all its properties for each driver instance found on the system. An add button  at the top of the list allows to add a new template, while deleting or editing a template is accessible via the contextual menu in the details area (vertical dots .



Media Types

The media types allow to define a global behavior for a set of devices, by defining on the one hand a general operation mode (the type of user who will use it as well as possible media templates and associated localization providers) and on the other hand the types of addresses associated with each driver instance: it is a matter of enumerating the combinations of message type and driver instances that can be used for this media type.

The administration screen displays on its left-hand side a list of existing types. Selecting one of these types displays in a detail area on the right all the combinations of message type and driver instance that are allowed. An add button  at the top of the list allows you to add a new type while deleting or editing each one is accessible via the contextual menu in the details area (vertical dots .



Each [Driver Instance / Message Type] combination can be renamed, modified or deleted by the icons at the end of the line ( and ), and other combinations can be added by the add button at the top of the list (.

Medias

The media represent all the devices allowing to reach a member of staff to requisition it, it can be for example mobile devices such as DECT phones or OfeliaSmart Android applications, or messaging applications installed on desktop computers, such as Ofelia Messenger.

The administration screen shows the list of media declared on the system on the left side of the screen, and a detail area showing the characteristics of the selected media on the right side. The list of media displays the name of the item, associated with an icon representing its availability status: online , offline  or not assigned to a user . An add button  at the top of the list lets the user declare a new media item, while deleting or editing each one is accessible via the contextual menu in the details area (vertical dots ).



When adding or modifying media, the following properties are required:

Nom *

1

Ce champ est obligatoire

Modèle de média *

2

3

☐ Média partagé

☐ Appliquer la configuration de périphérique du template

☐ Génération d'évènement système

- 1 **Unique Name** of the media
- 2 **Media type**: from the list of previously created types (see *Media Types* on page 117)

3 Options:

Shared Media: the media is not assigned to a single user; it is when he picks up the media from the rack that the user temporarily assigns it to himself

Apply configuration: see *Media Templates* on page 116

System event generation: should a failure create an event?

Once the main properties of the media have been set, the administrator can **1** assign it to a user, selecting the user and **2** indicating the priority that this media occupies among those already assigned to the user (Main media, first fold, second fold, etc.).

For each type of address **3** defined on the associated media type (see *Media Types* on page 117), it is then possible to define the address of the media for the corresponding type: this is the direct internal address **4** used to target the device on the driver, for example the telephone number for a DECT phone.

Notes:

- ⇒ The delete icon found at the end of the line allows to delete the address type of the media template and not the media itself. This operation is only possible if no media is already using this address.
- ⇒ It is not necessary to define an address for each type of address in the template, but only the addresses entered will be used to send a message when a specific type of address is required.

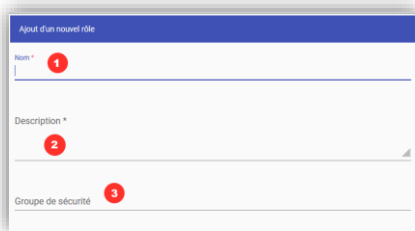
Roles and permissions

The different permissions that can be configured in the software can be finely defined by the administrator and assigned to a role. Each user can then be assigned one or more roles whose permissions add up and complete each other to define what the user can see and do in the whole software.

The role administration screen allows to create, delete and edit the different roles that can be assigned to users. It consists of a list of existing roles on the left and a detail area showing, when a role is selected in the list, all the permissions configured on it.

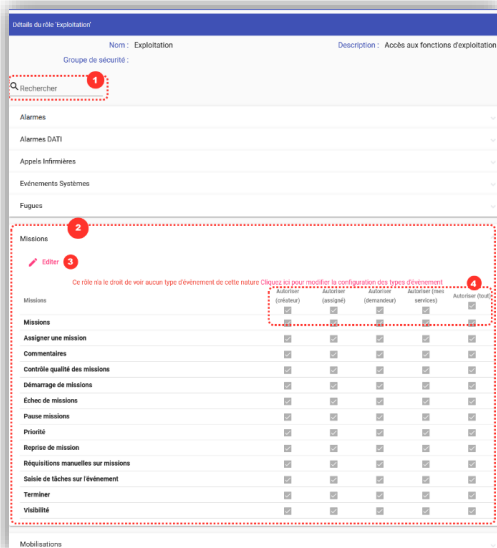
The '*Administrators*' role is built-in and cannot be removed. Each time the application is started and updated, a process ensures that the *Administrators* role has full privileges in the software and is limited only by the scope of the software license acquired from the site.

An add button at the top of the list  allows to declare a new role while the deletion or edition of each one is accessible through the contextual menu of the details area (vertical dots ). This same contextual menu also gives access to a '*Clone*' function which duplicates all the configuration of a role in order to use it as a template for a new role. The creation of a new role or its edition allows the administrator to fill in the following information:



- 1 **Unique Name** for the Role
- 2 **Description** of the role
- 3 **Security Group**: optional information to link it to a group in an external system (LDAP, Active Directory, etc.)

Once the general definition of the role is created, a list of all known permissions in the system appears, sorted by functional categories:



- 1 A search box allows you to find a permission by name and open the corresponding category.
- 2 Each category can be opened independently and reveal the permissions it contains
- 3 A button in the header of each category allows you to switch to edit mode.
- 4 In this mode, the administrator can check and uncheck each permission for the current role.
- 4 When the administrator presses the '*Validate*' button that replaced the '*Edit*' button at the time of editing, the changes made to the permissions of the current category are saved in the role.

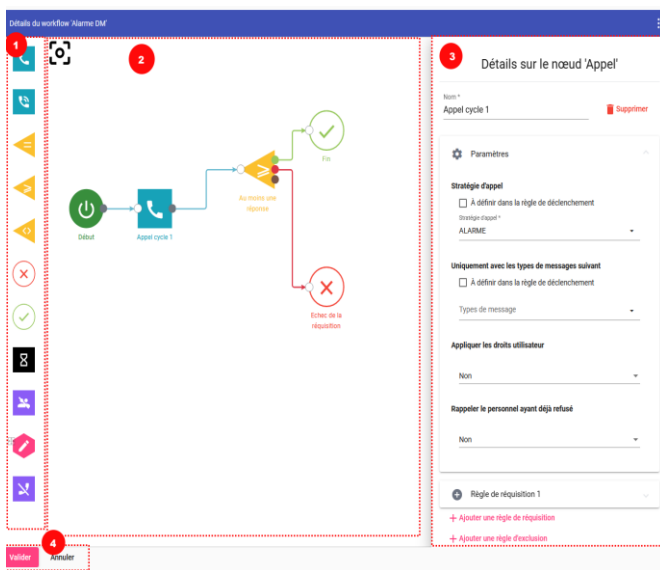
Notes:

- ⇒ Upon saving changes in a category, new permissions can be dynamically checked by the software in order to enforce functional requirements and to guarantee the consistency of the whole permission system. For example, if a role defines the permission to modify an employee, the members of this role will have to be able to browse the list of departments to which this employee can belong: for this reason, Ofelia® automatically adds the *Read* right on the list of departments as soon as the modification right of a user is granted. A set of dependency rules between permissions is thus validated by the system at each registration of a role.
- ⇒ The checkboxes describing the granularity of the permissions vary according to the permission edited, depending on the corresponding business rights. Details of the permissions available for each category and their granularity of rights are described in the *Permissions* chapter on page 52.

Workflows

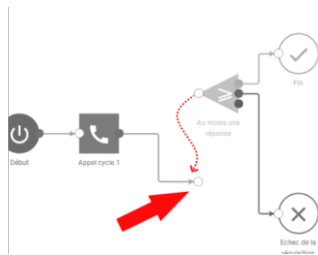
Workflows are used to describe the logic of the scenarios that will be executed during a mobilization or a notification. The administrator can graphically arrange different activities by linking them with branches according to their status (success, failure, timeout, etc.).

The administration screen is presented in the form of a list of scenarios on the left and a details area on the right. The latter shows a graphical display of the workflow and allows you to edit its content. As soon as a workflow is selected in the list and its content is drawn in the editing area, it is possible to act on its representation:



- 1 A **Tool palette** on the left-hand side of the drawing area contains a list of activities that the administrator can drag and drop onto the workflow being edited
- 2 The **drawing area** allows to move or link each activity using the mouse
- 3 Selecting an activity displays a sliding **configuration pane** on the right side of the screen. From this configuration pane, all the properties of the activity can be modified.
- 4 All changes made to the workflow are only saved after the user presses the '**Validate**' button.

The administrator can select the activities in the tool palette and drag them with the mouse to position them in the desired location on the work area. When hovering with the mouse over an activity already placed on the drawing, a cross appears allowing to delete the element.

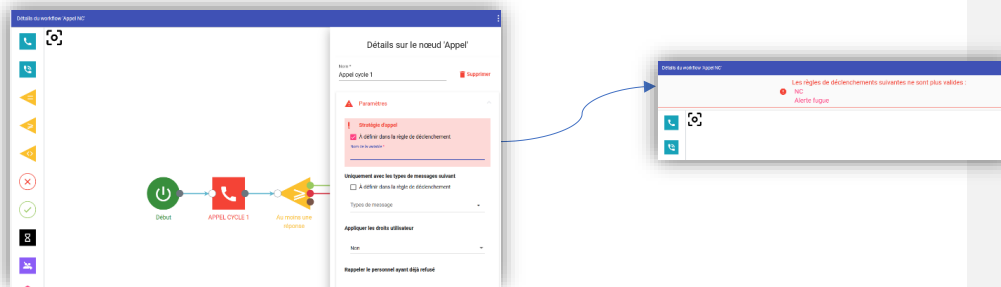


By selecting one of the anchor points between a connector and an activity, the administrator can:

- **Link** it to an activity or **change the target activity** to which the connector is linked by docking it to the input of another compatible activity
- **Remove the link** by releasing the connector outside an activity anchor area

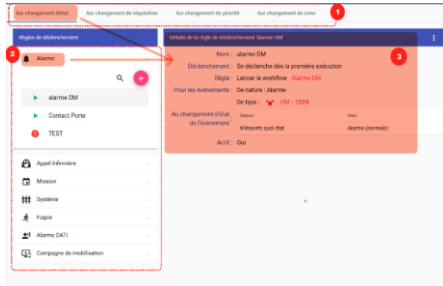
All available activities and their parameters are described in the chapter *Workflow activities* on page 27, as well as in the chapter *Workflow conditions* on page 30 for the condition nodes (logical branches). The specific parameterization of the requisition rules is described in the chapter *Requisition rules* on page 30.

If one of the parameters of an activity is defined on a variable, the engine automatically checks that the aforementioned variable is filled in all the triggering rules that target this workflow. If not, a warning is immediately displayed at the top of the drawing area to notify the administrator which rules he must complete: indeed, as long as the variables are not properly filled in, the rules are invalidated and paused. The requisition engine will therefore stop evaluating them as long as their content has not been completed.



Triggering Rules

Once the administrator has created the workflow definitions, he has to define which conditions will trigger their execution (see *Triggering rules engine* on page 32). This is done by creating appropriate triggering rules. The creation and editing of these rules are done via a dedicated screen which is presented as a double entry reading grid:



① On the top are **tabs** indicating the **triggering mode** of the rule.

② Depending on the tab selected, **the list of rules** corresponding to this triggering mode is displayed on the left of the screen.

③ The **detail and editing area** of the rule corresponds to the rule selected in the list, with the triggering mode chosen in the tab.

The creation of a new rule is done by means of the add

button  located at the top of each type of event, the edition of an existing rule is triggered by the contextual menu (vertical dots ). In creation mode, the administrator defines which workflow to launch and what the criteria are. All the criteria must be checked in order for the workflow to be executed, otherwise the rule engine moves on to the next one.

① **Unique name** of the rule, as it will appear in the workflow summaries.

② **Workflow** that should be started if all the conditions of the rule are met.

③ **Start State** in which the event was before the transition

④ **Arrival State** in which the event is at the end of the transition

Note: Only direct transitions from the start state to the end state are taken into account. However, it is possible to add several transitions {start state; end state} ⑤ to trigger the rule: any of these transitions is then valid for the rule.

⑥ **Active:** This checkbox allows to activate or suspend the evaluation of the rule. In the latter case, the engine will ignore the evaluation of the rule and go directly to the next rule.

⑦ **Only when the zone uses manual regulation:** If the control mode of the zone or the zone department is not set to "Manual Requisition", the rule is ignored. This allows to temporarily activate a set of rules according to the current mode of the department or the zone (for example to execute specific scenarios when no operator is present).

- 8 **Additional Criteria:** Allows to enter as many additional filter criteria as necessary to restrict the scope of the rule. Each additional criterion can be deleted with the trash bin icon at the end of the line (🗑).

Additional criteria include the following:

- Event Types:** List of event types on which to restrict the application of the rule. The concept of event type hierarchy (see *Event Types Hierarchy* on page 128) applies: a rule that applies to an event type will also apply to all its child event types.
If the event type has parameters, additional restrictions can be applied to these parameters:

These conditions are expressed in the form of a Boolean expression, for example

@Color == 'Red Code' OR @Temperature > 25

Pressing the @ key allows you to select one of the parameters of the selected type:

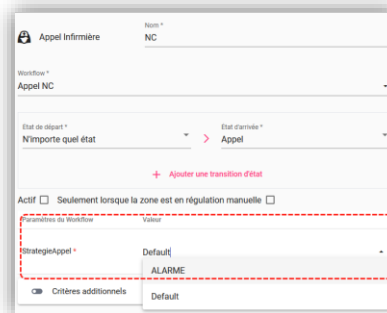
Moreover, the administrator can add several event types in this criterion: the rule will be valid if the event belongs to *any of the* specified types (assuming the parameters of these event types respect the stated conditions).

- Zones:** List of zones the rule application should be restricted on. As for the event types, the concept of hierarchical zones applies (see *Inheritance* on page 45): a rule for a floor, for example, will apply to all the rooms located on that floor.
- Zone Types:** List of zone types (e.g., room, hallway, floor, etc.) to which the rule applies exclusively.
- Tags:** The rule applies only to events located in a zone whose type includes one of the specified labels.
- Departments:** List of departments on which the rule applies: only events located in a zone belonging to one of these departments/units will trigger the rule.
- Priorities:** Priority list of events concerned by the rule. This filter allows for example to trigger a specific workflow only when an event is flagged as *urgent*.
- Beacons:** List of detection beacons (see *Beacons* on page 44) associated with the event. Thus, in the case of a runaway for example, a scenario can be started as soon as the fugitive passes in front of a particular sensor.
- Default job role:** Each type of event can be associated with a default function (possibly inherited from its parent). This filter allows the rule to be triggered only if the processed event has a type associated with one of the specified functions, for example only on missions processed by personnel with the '*Maintenance staff*' function.

- **Number of executions:** A workflow can be triggered consecutively several times for the same event, for example after a callback monitor has been activated (see *Reminder Monitors* on page 47). This filter allows the rule to start a different workflow after a certain number of executions.
- **Roles in the event:** List of User/Role combinations needed to trigger the rule. This is to restrict the rule to events for which named users occupy a specific role. Identifiable roles are *Creator*, *Requester* or *Recipient*: Thus, a rule may only apply if an assignment is created by Mr. X and destined for Ms. Y.

Workflow Variables

If activities of the selected workflow have deported some of their parameters in the form of variables, these variables must be entered in the rule, otherwise the evaluation of the rule will be suspended. Entering the workflow variables then becomes a mandatory field in the rule edition, as for example in the following screen where a call activity of the target workflow has defined a variable 'CallStrategy' as the requisitions' call strategy.



Order of precedence

When an event performs one of the transitions defined in the upper tabs (change of state, change of requisition state, change of priority, or change of zone), the list of corresponding rules for the nature of events is evaluated in the order of their display until one of them fulfills all the conditions. In this case, the rule engine stops evaluating the other rules and starts running the workflow specified in the rule.

Since only one rule can be triggered for each event transition, the evaluation order is of particular strategic importance. For this reason, by dragging and dropping the rules with the mouse, the administrator can order them so that the most restrictive rules are evaluated first and then the most general rules are evaluated last.

Call Strategies

The call strategy allows you to define how successive attempts should be made to reach users during requests. Once defined, they can be specified in the workflows or amended by the triggering rules if a variable is used.

The administration screen of these strategies displays the list of existing strategies on the left side of the screen, and a detail area with the characteristics of the selected item on the right side. An add button at the top of the list  allows to declare a new strategy, while the deletion or editing of each of them is accessible via the contextual menu in the details area (vertical dots ).

Création d'une nouvelle stratégie d'appel

Nom * 1
Ce champ est obligatoire

Nombre de tentatives * 2
1

Temps entre 2 tentatives (en sec) * 3
0

Profil d'appel * 4

- 1 **Unique Name** of the strategy
- 2 **Number of Attempts** after this number the call will be considered as definitively failed
- 3 **Time between 2 attempts** Waiting time between each new call, in case of failure (in seconds)
- 4 **Call Profile**: see *Call profile* on page 41.

Events

This sub-area contains all the elements necessary for the definition of event types and their behavior (monitors, actions, quality-of-service, associated message formats, etc.). The vertical menu gives access to the following features:

	System Event Types
	Nurse Call Event Types
	Runaway Event Types
	LWP Alarm Event Types
	Technical Alarm Event Types
	Mission Event Types
	Mobilization Event Types
	Mobilization Campaign Event Types
	Quality of Service Monitors
	Reminder Monitors
	Out-Of-Time Monitors
	Event State Definitions
	Event Priorities
	Forms Choice Lists
	Actions Lists
	Temporal Alarm Profiles
	Visit Reminders

Event Types Edition

The event type administration screens are accessible for each event nature via a separate menu for reasons of readability and practicality. However, their operation is completely identical whatever the nature of event concerned. The detailed operation of such a screen is illustrated here with the nurse call event types, but the exact same rules apply to all others (system events, missions, alarms, etc.).

The event type edition window presents **A** a tree structure of event types on the left side of the screen and **B** a detail area on the right side, displaying the attributes of the type currently selected in the tree and allowing its deletion or edition by means of a contextual menu (vertical dots **i**).



Event Types Hierarchy

The Event Type hierarchy tree allows you to add a child node at any level, move a node to another location in the tree, or delete a node. These operations are available through the appearance of icons next to the selected node: **+** to add a child and **↔** to move the node to another location. The root of the types of each nature must always be unique, allowing you to simply modify the behavior of an entire event nature by exploiting the notions of inheritance. For each nature, built-in event types, which can be modified in whole or in part, are provided with the software.

Each event type must have at least one parent (except for the root nodes of each nature), a unique name and a code. In addition to the main attributes of each event type, the administration screen allows you to directly specify the message format of each event type as well as the associated user permissions, through tabs **C** located in the detail area.

The hierarchy of event types by nature allows for inheritance of certain properties. For example, a priority group set on the parent root mission type will automatically be applied to all missions regardless of their child type. Similarly, if a characteristic such as the color or icon representing an event type is changed, the change will be immediately reflected in all child event types that have not overridden those properties themselves. This inheritance applies to all depth levels and can be modified for some children: each time a software property is inherited from one of its parents, the inheritance link can be broken thanks to the icon **🔗**: in our example, the administrator can thus break the inheritance of some mission types to assign them another icon despite what is specified at the higher level. On the other hand, the icon representing a chain (**🔗**) allows to recreate the inheritance link and to benefit again from the attributes of the parent on a property that had been modified.

Common Properties

Whatever the nature of the event, the following properties can be edited by the administrator:

- 1 **Unique Name** to identify the type of event
- 2 **Unique Code** that may appear, for example, in message formats for locations with limited display space
- 3 **Icon** to visually identify the type of event, to be chosen from a predefined list
- 4 **Display Color** (appears on dashboards and most event representations of this type)
- 5 **Format**: text and variables allowing to generate an event name (code) associated to this type during its creation
- 6 **Short Format**: text and variables allowing to generate a short name of event (code) associated to this type during its creation
- 7 **Expected Duration**: information value indicating the estimated duration of this type of event (in minutes)
- 8 **Expected Duration in case of a customer request**: information value indicating the estimated duration of this type of event (in minutes) if the request comes from a customer. This field is mainly used in the hotel industry.
- 9 **Maximum number of locations**: Allows the system to limit the route history kept in the system for an event (e.g., number of locations for a LWP alarm or a runaway)
- 10 **Minimum GPS change detection threshold**: required interval (in meters) between two GPS positions to detect a change. This allows you to adjust the sensitivity threshold of detections depending on the actual accuracy of each category of beacons.
- 11 **Additional Options**:
 - ⇒ **AutoStart**: mainly used in the context of missions, this flag lets an event automatically switch to started state immediately upon its creation

- ⇒ **Assignable to an event:** If set to 'false', this setting lets the administrator create "structural" event types, i.e., not directly linked to an event but rather used as a grouping for several child types. These non-assignable event types allow properties to be configured globally and then inherited by all children.
- ⇒ **Customer request checked by default:** UI behavior used for missions.
- ⇒ **Prohibit the creation of several active elements of this type in the same zone:** forces the uniqueness of the current elements in a zone. For example, this prevents two identical missions from being launched simultaneously in the same room.
- ⇒ **Mandatory Zone for creation:** some event types do not necessarily have a zone (for example missions). This option allows you to force the allocation of a zone to certain types of events.
- ⇒ **Hidden by default:** makes the event appear directly in a hidden way, i.e., in the 'hidden events' tab of the current events dashboard.

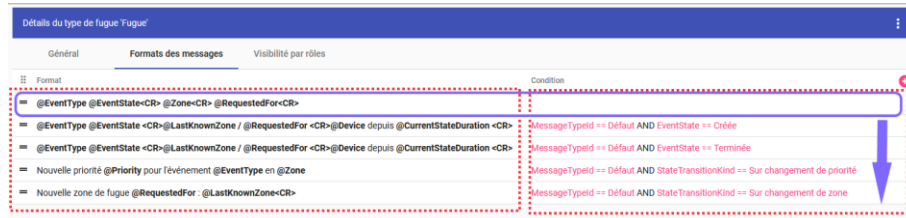
- 12 **Retention time before archiving:** if automatic purging is enabled, events of this type will be deleted from the database after the specified number of days. If nothing is specified, the events are kept indefinitely until a manual purge.
- 13 **Default job:** a job/function associated with this type of event. In the workflows requisition rules, the administrator can request the call of staff occupying the specified function at the time of the triggering.
- 14 **Event Priority Group:** each type of event can be associated with a priority group, which in turn gives access to different priority levels. (See *Event Priorities* on page 49)
- 15 **Default Event Priority:** within the previously defined group, the initial priority level of events of this type when they are created (for example: 'Urgent')
- 16 **Automatic start for zone types** instead of deciding that all events of this type start automatically, this option allows to specify that this behavior only takes place if the event zone has a particular type (for example, only in a room)
- 17 **Event Monitors** allows to associate a list of event monitors that will be activated when an event of this type meets the conditions defined in these monitors. Without this association, a monitor created in the program will never be triggered. (See *Editing Event Monitors* on page 135)
- 18 **Sound Profile:** allows you to associate a sound profile with all events of this type, indicating which sound to play for each transition state (see *Sound Profiles* on page 115).
- 19 **Actions list:** allows to select a list of actions (See *Actions and Forms* on page 49) which can be associated with events of this type (this can be the entry of a maintenance task list, taking a temperature, etc.)
- 20 **Parameters:** in addition to all the properties defined above, the administrator can add as many custom parameters as he wishes, in the form of a {Name/Description} list. These parameters can be marked as mandatory or not, and when necessary, the GUI will ask the user to fill in their value. The value of these parameters can then be used by the rules engine to decide which scenario to launch, and they can also be displayed in the message formats issued during the requisitions.

Message Formats

For each type of event, the administrator defines the format of the messages to be sent to the media, as well as the criteria for choosing the appropriate message format.

The writing of the message formats is based on a mixture of free text and keywords which are replaced at the time of sending the message by the current context of the event. Details of these formats are given in the chapter *Keywords and variables in messages* on page 131.

The message format edition tab is made up of lines with the message format in the first column, and the selection condition for the line in the second.



The very first line is fixed, cannot be deleted, and has no conditions: this is the format that will be applied if no other conditions are met.

The following lines can be edited or deleted using a contextual menu at the end of the line (✎ 🗑️). They allow alternative messages to be defined if the logical condition present in the first column is verified. An add button at the top of the list allows to add as many conditional format lines as necessary.

When a message is sent, the lines are analyzed in order, starting from the 2nd line. As soon as one of the conditions described in the first column is verified, the message format is applied. If not, the engine goes to the next line. If at the end of the list no condition has been met, the message format of the 1st line (the one without condition) is applied.

The logical conditions are a Boolean expression, i.e., a set of logical comparisons possibly combined by *OR* and *AND* operators. The contextual elements that can be used to write these Boolean expressions are the following:

Variable	Description
MessageTypeId	Type of message (see <i>Message Types</i> on page 84)
EventState	Current state of the event
PriorityId	Priority level of the event (see <i>Event priorities</i> on page 140)
RequisitionState	Current state of the Requisition
StateTransitionKind	Type of current transition of the event (for example: change of state, change of priority, etc.)
Zoneld	Name of the zone where the event is located

For each variable that requires it, a selection list appears when entering the interface.

Keywords and variables in messages

The Format and Short Format fields allow the use of keywords in the text entered. These are accessible by preceding them with the character '@' (at), and they are replaced by their actual value at the time of the creation of the event.

Keywords that can be used to define the event name generation:

Variable name	Content when replaced
Address	Media address
AutoBeacons	Last locations
BeaconDescription	Description of the current beacon
CreatedBy	Creator of the request
CurrentStateDuration	Elapsed time in this state
CustomerSatisfaction	Customer satisfaction Request (Yes/No)
Device	Device name
DevicePhoneNumber	Phone number of the Nurse Call device
DevicePosition	Position of the Nurse Call device in its zone
EventCode	Event identifier
EventComment	Text comment of the event
EventGenerator	Source Event generator name
EventShortDate	Event date (format DD/MM/AAAA)
EventShortTime	Event hour (format HH:MM)
EventState	Name of current event's state
EventType	Name of event's type
EventTypeCode	Code for event's type
ExpectedDuration	Expected duration of the event
LastKnownZone	Name of the last known location for the event
LastKnownZoneCode	Code of the last known location of the event
Name	Event full name
NbExecutions	Number of times an associated workflow has been executed for this event in this state
ParentMessage	Message of the event's parent (e.g., in a Mobilization Campaign)
Priority	Name of the current priority of the event

PriorityValue	Value of the current priority of the event (numeric)
RequestedBy	Name of the person who made the request
RequestedFor	Recipient of the request (for example for a mission, this could be the customer)
StaffMembers	List of users requested for the event (comma separated)
StateChangedBy	Name of the person who initiated the state change
TriggerMode	Creation mode (manual / automatic)
UserPreferredLanguage	Customer's language
Variable	See explanations below
Zone	Name of the zone where the event is occurring
ZoneCameraUrl	URL to access a video stream of the event zone
ZoneCode	Code of the zone where the event occurred
ZonePath	Path to the zone in the hierarchy (« breadcrumb » like 'North Aisle-Floor 3- East Corridor- Room 314')

Some keywords also allow you to specify an additional format or parameter: for example, when using the **@Variable** keyword, the editing popup window that appears when you click on the keyword allows you to choose the name of the system variable that you wish to replace at the place where the term **@Value** appears.

Options de la variable

Format

valeur de @Value

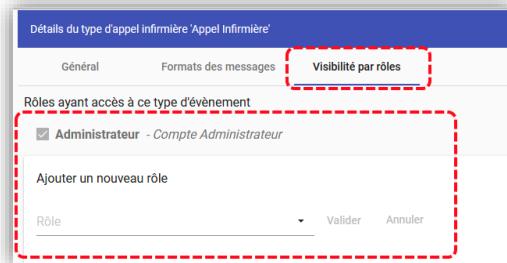
Insérer "@value" qui sera remplacé par la variable

Variable *

Valider Annuler

Visibility for each role

A last tab in the detail area of an event type allows you to define which security roles are authorized to view this type of event (see *Roles and permissions* on page 119). The administrator can thus add or remove a role from the list of those authorized for the type of event, thus adjusting security as closely as possible according to the profile of the application's users.



Détails du type d'appel infirmière 'Appel Infirmière'

Général Formats des messages **Visibilité par rôles**

Rôles ayant accès à ce type d'évènement

☒	Administrateur - Compte Administrateur
-------------------------------------	--

Ajouter un nouveau rôle

Rôle Valider Annuler

To do this, simply select a new role from a list, or uncheck one of the roles already authorized.

Editing Event Monitors

Quality-of-Service Monitors

Quality of Service monitors are used to measure the time elapsed between two transitions of an event and assign them a quality level. The administrator defines several time steps and defines a name and a color for each one of them. As an event evolves, as time elapses, it gets classified into the corresponding level. This level can then be displayed in the dashboards (in real time) or used retrospectively for statistical analysis.

The Quality-of-Service monitors screen shows a list of existing monitors on the left side of the screen, and a detail area showing the characteristics of the selected monitor on the right. An add button at the top of the list  is used to declare a new monitor, while the deletion or editing of each monitor is accessible via the contextual menu in the details area (vertical dots ).

- 1 **Icon** (picked from a predefined list)
- 2 **Unique Name** of the monitor
- 3 **Nature of the associated event** (Mission, Alarm, etc.)
- 4 **Type of triggering:**
 - o **Following a successful or failed requisition:** the timing starts as soon as a requisition workflow is completed, regardless of its result
 - o **Following a successful requisition:** the timing starts as soon as a requisition workflow results in a Success
 - o **Following a failed requisition:** the timing starts as soon as a requisition workflow fails (i.e., lands in a Failure node)
 - o **Immediately:** timing starts as soon as the transition between the start and finish states occurs.
- 5 **Start state:** State in which the event was at the moment the timing started.
- 6 **Arrival State:** State in which the event is at the end of the transition.

Notes regarding states:

- The start state or the finish state can be multiple, but not both at the same time. For example, one can define a quality monitor on nurse calls that measures the time elapsed between the call state and the presence or end state, or a monitor that times between the call or emergency state and the presence state, but it is not possible to define a monitor that would both monitor time between the call or emergency state *and* the presence or end state.
- The monitored states are not necessarily consecutive: if a QoS monitor measures the time elapsed between a Call state and an End state, the event can have a cycle that makes it successively cross the *Call*, then *Emergency*, then *Presence* and *Assistance* states before finally reaching the *End* state. It is then the total time between the two measured states (between Call and End) that will be timed, whatever the intermediary steps.

7 Main Quality Control It is possible to associate several monitors with the same type of event but, for reasons of legibility, the dashboards only allow the display of one monitor per type of event. This checkbox allows you to indicate which monitor should be displayed in the summary lists. Of course, accessing the details of an event will display the details of all the associated monitors.

8 When a new Quality-of-Service monitor is created, a built-in quality level is already created: the **overflow level**. This level cannot be deleted, it is the time beyond which the quality is considered in its most degraded state. The configuration therefore consists of defining the characteristics of this maximum state, and then creating any intermediate time slots.

The addition of a new intermediate level is done by using the dedicated add button and then by entering a **color**⁹, a **name**¹⁰ and a **duration**¹¹ corresponding to the added quality level.

Note: Once a monitor has been created, it is necessary to associate it with the event types that will use it.

Reminder Monitors

The purpose of reminder monitors is to restart a state transition of an event if a target state has not been reached after a predefined duration. For example, it can be defined that if a nurse call is triggered and after 10 minutes no presence has been achieved in the room, the call state should be triggered again (causing the associated requisition scenarios to run again).

The reminder monitor screen has a list of existing monitors on the left side of the screen, and a detail area with the characteristics of the selected monitor on the right. An add button at the top of the list ⁺ allows to declare a new monitor, while deleting or editing each one is accessible via the contextual menu in the details area (vertical dots ⁱ).

- 1 **Icon** (picked from a predefined list)
- 2 **Unique Name** of the monitor
- 3 **Nature of the associated event** (Mission, Alarm, etc.)
- 4 **Type of triggering:**
 - o **Following a successful or failed requisition:** the timing starts as soon as a requisition workflow is completed, regardless of its result
 - o **Following a successful requisition:** the timing starts as soon as a requisition workflow results in a Success
 - o **Following a failed requisition:** the timing starts as soon as a requisition workflow fails (i.e., lands in a Failure node)
 - o **Immediately:** timing starts as soon as the transition between the start and finish states occurs.
- 5 **Start state:** State in which the event was at the moment the timing started.
- 6 **Arrival State:** State in which the event is at the end of the transition.

Notes regarding states:

- The start state or the finish state can be multiple, but not both at the same time. For example, one can define a monitor on nurse calls that measures the time elapsed between the call state and the presence or end state, or a monitor that times between the call or emergency state and the presence state, but it is not possible to define a monitor that would monitor between the call or emergency state *and* the presence or end state.
 - The monitored states are not necessarily consecutive: if a monitor measures the time elapsed between a Call state and an End state, the event can have a cycle that makes it successively cross the *Call*, then *Emergency*, then *Presence* and *Assistance* states before finally reaching the *End* state. It is then the total time between the two measured states (between Call and End) that will be timed, whatever the intermediary steps.
- 7 **Maximum duration:** Time beyond which the reminder will activate and force the new transition on the event.

Out-Of-Time

Out-of-time monitors are a specific feature of Mission events. These monitors are very similar to the reminder monitors, and allow two additional actions: forcing a mission failure when the monitor's time limit has expired, and possibly launching a new event (of any kind).

Apart from these two additional options, the Out-Of-Time Monitor screen is similar in all respects to the Reminder Monitor screen (see chapter above). It shows a list of existing monitors on the left-hand side of the screen, as well as a detail area showing the characteristics of the selected monitor on the right-hand side. An add button at the top of the list  is used to declare a new monitor, while the deletion or editing of each monitor is accessible via the contextual menu in the details area (vertical dots ).

1 **Icon** (picked from a predefined list)

2 **Unique Name** of the monitor

4 **Type of triggering:**

- **Following a successful or failed requisition:** the timing starts as soon as a requisition workflow is completed, regardless of its result
- **Following a successful requisition:** the timing starts as soon as a requisition workflow results in a Success
- **Following a failed requisition:** the timing starts as soon as a requisition workflow fails (i.e., lands in a Failure node)
- **Immediately:** timing starts as soon as the transition between the start and finish states occurs.

4 **Start state:** State in which the event was at the moment the timing started.

5 **Arrival state:** State that triggers the cancellation of the reminder monitor.

6 **Maximum duration:** Delay after which the reminder will activate and force the new transition on the event.

7 **Force failure on expiry:** If this box is checked and the time limit is reached without the event having passed into one of the defined arrival states, the mission is automatically switched to the failure state ("Could not complete").

- 8 **Type of event created in case of failure:** If an event type is selected in this list and the deadline is reached without the event having entered one of the defined arrival states, a new event of the defined type is automatically created.

Notes regarding states:

- The start state or the finish state can be multiple, but not both at the same time. For example, one can define a monitor on nurse calls that measures the time elapsed between the call state and the presence or end state, or a monitor that times between the call or emergency state and the presence state, but it is not possible to define a monitor that would monitor between the call or emergency state *and* the presence or end state.
- The monitored states are not necessarily consecutive: if a monitor measures the time elapsed between a Start state and a Completed state of a mission, the event can have a cycle that makes it successively cross the *Pause* then *Resume* states multiple times before finally reaching the *Completed* state. It is then the total time between the two measured states (between Start and Completed) that will be timed, whatever the intermediary steps.

Event state definitions

Each event type defined in Ofelia has a predefined number of states. Each of these states has a name (for example '*In progress*', '*Completed*', '*Suspended*', etc.) and a color that provides relevant information on the graphical representations of the views by department or by floor map.

The hierarchical structure of the underlying representations (for example the hierarchy of the zones) imposes to define for all these states a notion of priority and precedence. Indeed, on a view of a building for example, events coming from different levels will be overlayed one on top of each other: a system event on a floor for example, a fire alarm in a corridor, nurse calls in rooms, etc. When the whole building is displayed, you may wish to make the fire alarm appear in red as a priority, rather than the visual indicating that a mission is suspended.

To do this, the software allows administrators, according to the actual operational priorities defined on the site, to order all the event states, whatever their nature, and to edit the name and the color used to represent these states.

Nature de l'événement	État	Couleur	Applicable sur la carte
Système	Actif	#01F349	☒
Alarme D&T1	Volontaire 2	Couleur	☒
Alarme D&T1	Volontaire	Couleur	☒
Alarme D&T1	Arrachement	Couleur	☒
Alarme D&T1	Perte de verticalité	Couleur	☒
Alarme D&T1	Immobilité	Couleur	☒
Alarme	Alarme (très haute)	Couleur	☒
Alarme	Alarme (haute)	Couleur	☒
Alarme	Alarme (normale)	Couleur	☒
Alarme	Alarme (basse)	Couleur	☒
Alarme	Alarme (très basse)	Couleur	☒
Fugue	Crée	Couleur	☒
Appel infirmière	Urgence	Couleur	☒
Appel infirmière	Assistance	Couleur	☒
Appel infirmière	Appel	Couleur	☒
Appel infirmière	Rappel de visite	Couleur	☒
Mission	parti	Couleur	☐
Mission	En cours	Couleur	☐
Mission	Suspendue	Couleur	☐
Alarme D&T1	Alarme Temporelle 1	Couleur	☒

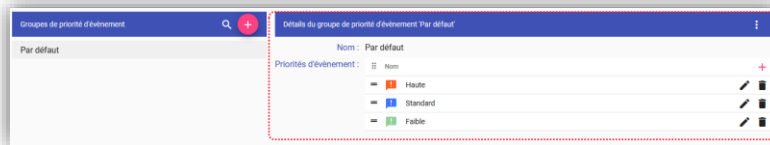
The state administration screen fulfills this role: it allows to reorder each state by drag-and-dropping so as to place the most important states at the top of the list, and thanks to the edit icon (✎), to rename the label of this state and adjust its color. Finally, a check box allows to define whether the chosen color should be applied in maps to represent the event: in this case, the event will appear with the color corresponding to its current state rather than to its underlying type (see *Event Types Edition* on page 128).

Event priorities

The event priorities are organized in priority groups, each grouping a coherent set of priorities (for example '*Urgent*', '*Standard*', '*Low*'). The purpose of these groups is to be able to assign a granularity and distinct label depending on the types of events occurring in the system.

The administration screen of the priority groups presents the list of existing groups on the left side of the screen, and a detail area presenting the selected group on the right. An add button (+) at the top of the list

allows to declare a new group, while deleting or editing a group is accessible via the contextual menu in the details area (vertical dots ). Creating and editing a group simply consists in giving it a unique name.



Once a group has been created, the administrator can assign different levels of priorities to it: a new priority level can be added using the add button located in the detail area, specifying a name, an icon and a color, and each priority level can be deleted or edited using the buttons placed in front of each ( ).

Finally, each item of the group has to be sorted in descending order of priority so that the rules to be applied according to the availability of each user can be defined later when an event of each priority level occurs. This sorting is done by simply dragging and dropping the priorities in the list.



Once a priority group has been defined by the administrator, it can be assigned to event types (see *Event Types Edition* on page 128). The configured group will then be inherited by all downstream event types.

Forms (Lists of Actions)

During the different stages of an event's life, an operator may have to collect information depending on the type of event being processed. For example, it may be necessary to take biological values during a treatment (temperature, blood pressure, etc.) or to carry out a series of maintenance operations during a mission in an industrial context (compliance measurements, checklists, etc.).

In order to unify this process and to keep up-to-date traceability of the information entered, all data collected is stored in the form of forms and directly associated with the corresponding event transition (triggering of an alarm, presence in a room, etc.). The administrator defines structured forms, grouping the actions to be filled in different folders, and decides on the restrictions to be applied: on which type of event, during which state transition, for which type of user function as well as the optional or mandatory nature of each question.

This form design is done through a specific administration screen. This screen presents a list of forms (also called '*Action Lists*') on the left side of the screen and a detail area on the right side for each selected item. An add button at the top of the list  allows you to declare a new action list while deleting or editing an existing form is done via a contextual menu at the top right (vertical dots .

To create a new form, the administrator must first fill in the following information:

Création d'une nouvelle liste d'actions

Nom * 1

Données biologiques

Type de formulaire * 2

Formulaire d'événement

Nature de l'événement * 3

Appel infirmière

Services

Cardiologie × Traumatologie × 4

* Champs requis

Valider Annuler

- 1 **Unique Name** of the action list
- 2 **Form Type**: In the current version, only event forms are available. In the future, forms can also be free, or attached to zones or stays.
- 3 **Nature of the event**: In the case of a form of type *Event Form*, the nature of the events to which the instances of this form will be attached is required.
- 4 **Departments**: Optional list of units allowing to restrict the cases where this form will be usable, depending on the event zone. For example, a different form can be defined for the Pediatrics and Ambulatory Surgery departments.

Once these global settings have been defined, the administrator can add folders and actions within these folders. The folder tree is not limited in size and allows for an orderly and logical arrangement of numerous forms. For a simple use case, only one single folder is usually needed.

Détails de la liste d'actions Maintenance

Nom : Maintenance Type de formulaire : Formulaire d'événement

Nature de l'évènement : Mission

Services : Cardiologie
Pédiatrie
Traumatologie
Unité de Vie Protégée

Ajouter un dossier Ajouter une action

Ventilation VMC Climatisation Débit Température

Ajouter un dossier Ajouter une action Editer Supprimer

Nom : Température eau
seule fois par transition : Non
Restreint pour les états suivants :
Aide à la saisie : Gardiens
Code analytique : Température ballon
Type de valeur : Valeur numérique
Format : (Valeur) *
Unité de la valeur : °
Format pour l'export : @Value export
Liste de choix :
Demande validation : Non
Obligatoire : Non
Déclencheur de transition : Non

Editer Supprimer

Adding a folder or an action is done either by the dedicated buttons **1**, or by the contextual menu **2** that appears when the mouse passes over each element (folder or action) of the tree structure (**3**).

This contextual menu, as well as the buttons located at the bottom of the detail of the selected item, allow you to edit the contents of the folder or the action or to delete it.

Editing or creating a folder consists only in providing it with a name, while editing an action requires the following parameters:

The screenshot shows a form titled "Ajout d'une nouvelle action". It contains several fields with red numbered annotations:

- 1: Icone (Icon)
- 2: Nom * (Name)
- 3: Type de format * (Format type)
- 4: Unité de la valeur (Value unit)
- 5: Aide à la saisie (Input hint)
- 6: Format de la description (Description format)
- 7: Format pour l'export (Format for export)
- 8: Fonctions (Functions)

 There are also labels like "Ce champ est obligatoire" (This field is mandatory) and "Insérer 'l'variable' qui sera remplacé par la variable" (Insert 'the variable' which will be replaced by the variable). A footer note says "* Champs requis" (Required fields).

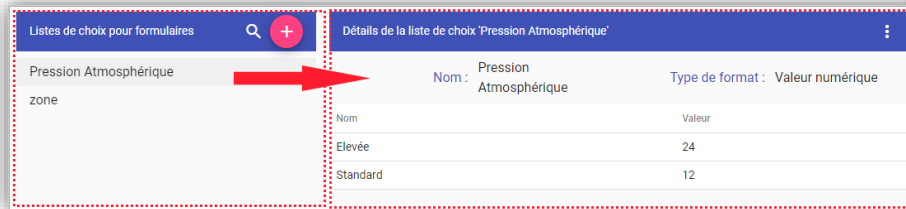
- 1 **Icon** representing the action, to be chosen from a predefined list
- 2 **Name** of the action, as it will be displayed to the user
- 3 **Format Type:**
 - **None:** for a checklist to be checked, for example for an instruction
 - **Numerical value:** for an integer or a number with a decimal point
 - **Text:** for any data containing alphanumerical characters
 - **Zone:** to allow the selection of a zone of the establishment (*Not used for the moment*)
- 4 **Value Unit:** Unit for physical measurements (for example, Kg, cm, °, etc.)
- 5 **Input Hint:** Explanatory text that will be displayed to the operator when entering a value
- 6 **Description Format:** Allows to format the value that has been entered to present it on the screen. The alias **@value** contains the value entered by the operator. For example, for a speed you can choose a format **@value km/h**.
- 7 **Format for export:** On the same principle as the description format, allows to format the value as it will be exported to a third-party system (for example, **@value cm**)
- 8 **Functions/Jobs:** Restriction on the functions of the users who can access this action. Thus, in the same form, it will be possible to reserve certain actions for doctors or to enter certain measures only for nursing staff.

Choice Lists for Forms

To simplify and guarantee the consistency of the entries of certain action values, the administrator can make available pre-constructed choice lists: this involves forcing the user to use a pre-defined choice instead of manually entering a value. This is particularly useful when the expected answers are part of a finite and small set.

Each list is named (e.g., "Color") and has a type (numeric or text values). Each item of the list is identified by the name that will appear to the user ("Red", "Green", etc.) and contains a value (whose type is defined at the list level) which will be substituted for the element chosen in the form (for example Red=> 1, Green =>2, etc.).

The choice list management screen allows to select a choice list on the left side of the screen. This list then appears in a detail zone on its right. An add button at the top of the list  allows you to declare a new choice list while the deletion or editing of an existing list is done via a contextual menu at the top right (vertical dots ).



The creation of a list consists first of all in defining its name and the type of data it will hold (Numeric or Text), then in creating one by one each item of the list by specifying their label and their value. A created item can be directly edited, or deleted via the icon located at the end of the line.



The choices created in this way are barely an input convenience and the values are not linked to the forms generated. It is therefore only a question of offering a simple and unified input mode for the user, and the deletion or editing of an entry in the list of choices will have no consequence on the forms already entered (no correction of the values already created for example).

Temporal Alarms Profiles

The temporal alarm profiles define a set of parameters used to manage the activation and behavior of a lone worker protection alarm when the user is in a non-coverage zone or during a trip (see *Temporal alarms* on page 58). When using a compatible device (*OfeliaSmart* for example), the user can activate one of the temporal alarms associated with a function that he is currently occupying (depending on his schedule, see *Assigning a Job to the user* on page 105).

The administration screen is divided into a list of profiles (left part of the screen), and a detail area presenting the selected profile, on the right. An add button at the top of the list  allows you to declare a new profile, while deleting or editing an existing profile is accessible from within the contextual menu in the details area (vertical dots .

The creation or edition of a time alarm profile consists in filling in the following properties:

- 1 **Unique Name** of the profile
- 2 **Disable media's system alarm**: If this box is checked, the loss of connection of the media will not cause the creation of a system event, even if this creation is specified for the media. This avoids the creation of redundant system events when the user enters an area out of coverage for example: the event will be replaced by the triggering of an alarm if the user does not come back within the time limit.
- 3 Duration definition mode: **by duration** (indicating a default number of minutes) or **by steps** (indicating the duration of a step 4 and the default number of steps 5).
- 6 **Arrival state** to be assigned to the event triggered if the alarm reaches its end. It can be one of the two built-in LWP Alarm states: either *Temporal Alarm 1* or *Temporal Alarm 2*.
- 7 **Pre-trigger reminder**: Before the alarm is actually triggered, the administrator can issue a reminder to the user in order to extend the delay or discard the alarm (possibly by leaving the non-coverage zone if necessary).
- 8 **Subsequent reminder**: if the reminder is triggered, the user can set a second reminder before the final launch.
- 9 **Restricted to functions**: the time alarm can be restricted only to employees assigned to one of the functions/jobs in this list.
- 10 **Default Instructions**: this is the message that will be displayed when the alarm is activated.

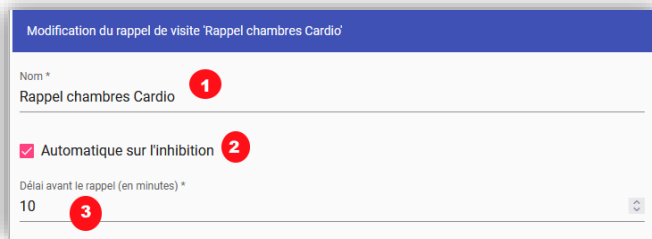
Visit Reminders

Visit reminders are a specific form of room monitoring, related to the issue of inhibited nurse calls devices (see *Visit reminders* on page 48). The visit reminder is a timing system linked to a department, which aims to impose that an attendance be made in the rooms of this department at regular intervals. It can be set up manually, or automatically as soon as a room's nurse call device is disabled.

The administrator sets the visit reminders before assigning them to the facility's departments (see *Departments/Units* on page 109) or directly to the nurse call boxes (see *Nurse Call Devices* on page 153).

As with most administration screens, the visit reminder screen consists of a list of reminders (left side of the screen), and a detail area showing the selected item, on the right. An add button  at the top of the list allows you to declare a new reminder, while deleting or editing an existing reminder is accessible via the contextual menu in the details area (vertical dots ).

The creation or edition of a visit reminder policy consists in filling in the following attributes:



Modification du rappel de visite 'Rappel chambres Cardio'

Nom * **1**
Rappel chambres Cardio

☒ Automatique sur l'inhibition **2**

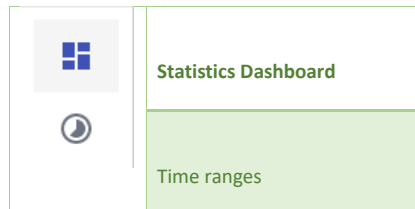
Délai avant le rappel (en minutes) * **3**
10

- 1 Unique name** of the reminder
- 2 Automatic upon inhibition:** If this box is checked, the reminder will be activated as soon as a nurse call box in the zone is inhibited by a staff member. Otherwise, the reminder will have to be activated manually.
- 3 Time before reminder:** Time in minutes between the activation and the creation of a nurse call in the 'Visit due' state if the room has not received any presence state.

Each time the room is in the presence state, the timer is reset, so if there are regular attendances within the time limit, the callback will never be triggered.

Statistical Analysis

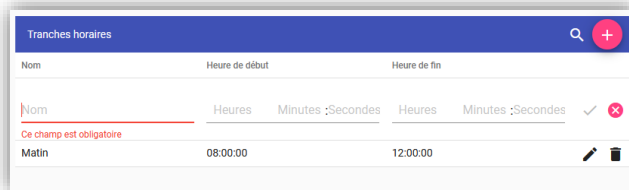
This sub-area allows the administrator to configure the statistics module integrated in Ofelia®. The vertical menu gives access to the following features:



Time Ranges

Time ranges are a way for the administrator to group the different time-stamped values of the system into meaningful time slots that are relevant for analysis. They can be time slots corresponding to the organization of work in the facility ('Morning', 'Afternoon', 'Night') or any other division useful for the analysis of past events. These time ranges can be used as a grouping level in the statistics.

The administration of the list of time ranges is done directly in the list by adding, deleting or editing elements:



The addition is done by the dedicated button located at the top of the list , while the editing or deletion is done by the contextual icons placed next to each item ( ). Each range consists of a name and a start and end time.

For reasons of consistency in the statistics displayed, the time ranges have to be both separate and exclusive: under no circumstances can two slots overlap (otherwise, the same event could be counted several times).

Statistics Dashboards

By taking into account the authorization profile of each user, the administrator can create personalized statistical dashboards displaying key indicators of the institution's activity. For each dashboard created, the administrator must indicate a name and a scope (i.e., the list of roles authorized to consult it), and then arrange the pre-configured graphic elements on a grid.

When a user accesses the statistics module (see *Statistics* on page 213), he/she is presented with a list of dashboards to which he/she has access, in an order defined by the administrator. Selecting one of these

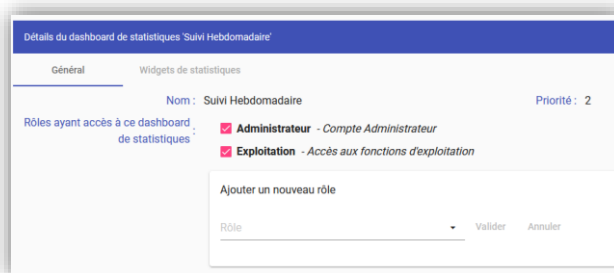
dashboards presents the recalculated indicators for that particular user, applying any additional restrictions that are specific to that user (for example, if the user cannot see all event types or if his or her authorization is restricted to certain particular departments).

As for most of the administration screens, this one is made of a list of dashboards (left part of the screen), and a detail zone presenting the selected element, on the right. An add button at the top of the list  allows the administrator to declare a new dashboard, while deleting or editing an existing dashboard is accessible via the contextual menu in the details area (vertical dots .

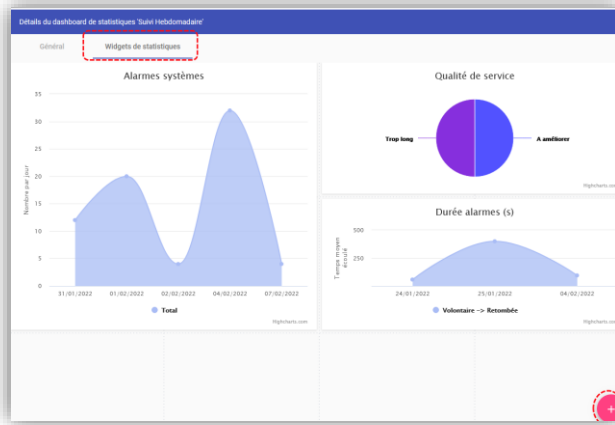
The creation is done in two steps: The administrator starts by naming the dashboard, giving it a display order (if the user has access to several dashboards, they will appear in different tabs of the statistics module according to this order), and defining the number of columns of which the working grid is made up.



Once this definition has been made, the administrator decides which roles are authorized to display this dashboard: Users belonging to these roles (see *Roles and permissions* on page 119) will automatically see the dashboard appear in their statistics module. Adding a new authorized role is done by selecting it in the drop-down list, while revoking the right to a role is done by unchecking it:



When the general properties are correctly filled in, a tab gives access to the configuration of the dashboard itself.



The screen then shows a grid (with the number of columns specified in the General tab) and an add button to position a new graphic element (aka 'widget').

Widgets

Adding a widget to the dashboard is done by following a three-step wizard:

Choosing a type of analysis

This step consists of giving the widget a title, and then choosing what will be calculated by the statistic: it can be a number of occurrences, a transition duration, or a distribution of QoS levels.

This step also allows you to choose the period over which the analysis should be performed: it can be either a *fixed* date range or a *dynamic* date range. The latter mode allows you to have a sliding date window that will be valid at any point in time (for example "last week" or "current quarter"). To use a dynamic date range, the administrator sets the unit of measurement (days, weeks, months, etc.) and indicates how far back in time from the current date the range starts, as well as the duration of the range.

For example, the following definition allows you to display the current week:

Unité de temps de départ *	Décalage par rapport à la date courante *	Unité de temps de l'intervalle *	Durée *
Semaine calendaire	1	Semaine	1

⇒ From the date of the report execution, go back one calendar week (i.e., positioning to the previous Monday) and then take a duration of one week.

To select the previous working week, from Monday to Friday, the definition would change on the length of the interval:

Unité de temps de départ *	Décalage par rapport à la date courante *	Unité de temps de l'intervalle *	Durée *
Semaine calendaire	1	Journée	5

Definition of additional filters

Once the calendar scope of the statistic and the type of counted objects has been set, the next step is to refine the filter on the analyzed events.

The only mandatory filter is the nature of events (because of their specific status, different natures of events cannot be mixed within the same graph). The set of possible filters is given below:

- 1 **Nature** of the events (Missions, Alarms, Nurse Calls, etc.).
- 2 **Initial state**: Only events that had this state at the time of their creation.
- 3 **Visited state**: Only events which, in the course of their life, have passed through the indicated state at some point.
- 4 **Event Types** concerned (multiple choice among the types of events of the selected nature). Note that inheritance applies and that the selection of a parent type automatically implies that of its children.
- 5 **Units** (multiple choices): these are the departments (medical units) to which the zones where the events occur are attached.
- 6 **Zone types** of the events (for instance, only events located in a room or in a common area).
- 7 **Zones** (multiple choices): Only events that took place in one of the indicated zones. This can be a leaf level (e.g., a room) or an upper level grouping all its descendants (for example, a floor grouping all the corridors and rooms).
- 8 **Default job**: Only events for which the default function matches.
- 9 **Staff Member**: limits the selection of events to those requested by a particular employee.
- 10 **Time Range**: Only the events *created* during the custom time range indicated (see *Time Ranges* on page 147).
- 11 **Quality-Of-Service**: Only the events corresponding to a specified quality of service.
- 12 **Levels**: Only the events that correspond to the selected quality levels (among those of the QoS previously selected).

Choosing a grouping and aggregation mode

The last step in defining the widget is to indicate how the selected items should be grouped (by date, by day, by department, etc.) and according to what type of aggregation (total number, average, etc.).

The combination of possible choices depends on the type of analysis chosen in the first step of the wizard. The following table summarizes the combinations:

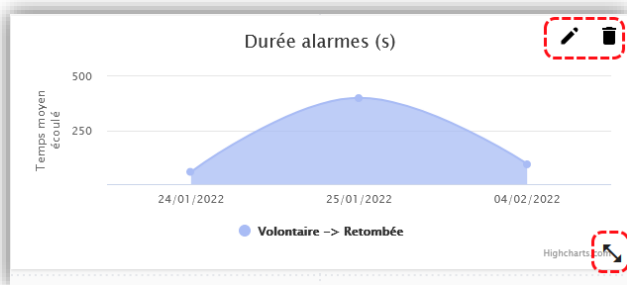
Analysis Type	Grouping	Aggregate
Number of occurrences	Date (for each date of the interval)	Total Amount
	Week	Total Amount
		Average per hour interval
		Average per day
	Hourly Interval	Total Amount
		Average per hour interval
	Event type	Total Amount
		Average per day
		Average per week
		Average per month
	Custom time range (*)	Total Amount
		Average per day
	Date and custom time range(*)	Total Amount
Quality-of-Service	Date	Total Amount
	Week	Total Amount
		Average per day
	Hourly Interval	Total Amount
		Average per day

Transition Duration	Average per week	
	Average per month	
	QoS Level	Total Amount
	Date	Total Amount
	Week	Total Amount
		Moyenne par jour
	Hourly interval	Total Amount
	Custom Time Range (*)	Total Amount
		Average per day
		Average per week
		Average per month
	Date and custom time range (*)	Total Amount

(*) for example, 'Morning' or 'Afternoon', depending on the time slots created by the administrator (See *Time Ranges* on page 147)

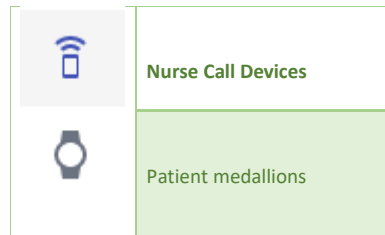
Graphic layout

When the widget is configured, it appears in a free space on the dashboard and occupies by default a single column of the grid. Icons at the top right allow you to launch the wizard again to modify the configuration, and an icon in the bottom-right corner (*) allows you to grab the widget to move it on the grid or change its size and make it occupy a different number of "slots" on the dashboard.



Devices

This sub-area allows you to manage Ofelia's® integrated devices such as nurse call boxes and patient medallions (call and escape). The vertical menu gives access to the following features:



Nurse Call Devices

The nurse call devices administration screen allows the administrator to manage the list of all the boxes known in the facility and to edit their properties (address, behavior) or location (room, position in the room).

The screen is typically made up of a list of devices (left part of the screen), and a detail area on the right presenting the characteristics of the selected box. An add button  at the top of the list allows you to declare a new box, while the deletion or editing of an existing device is accessible via the contextual menu in the details area (vertical dots ). Editing the properties of a device consists in filling in the following information:

- 1 Unique Name** of the device (mandatory), used to identify it throughout the software
- 2 Phone Number:** optional number to communicate with the unit, if it is equipped with a phone grid.
- 3 Zone** (mandatory): name of the zone in which the box is located (usually a room name)
- 4 Position type** inside the room (bed, door, toilet, etc.) as defined in chapter *Zone Types* on page 95.
- 5 Identifiant** (mandatory): unique address of the device that can be controlled by the driver instance. It is a manufacturer address which allows the equipment to be uniquely addressed in the facility.

- 6 **Driver Instance:** Driver instance in charge of the communication with this device (see *Drivers and driver instances* on page 22).
- 7 **Nurse Call Type:** Type of event to be created when this device is activated, among the types of nurse call events defined by the administrator (See *Event Types Edition* on page 128).
- 8 **Visit Reminder:** optional visit reminder strategy associated with the box (See *Visit Reminders* on page 145). The icon picturing a chain  is used to indicate whether the callback is inherited by the service and, if so, to "break" this inheritance. This option is useful if a whole department must be protected by a reminder except for some specific devices.
- 9 **System Event Generation:** if this box is checked, a communication failure between the driver and the department will trigger the creation of a new system event.
- 10 **Controllable Nurse call device:** if a driver instance supports it, it allows to control the nurse call box to turn on the viewport for example by slaving it to Ofelia®. If this box is checked, it is necessary to indicate the driver instance  that supports this operation and the controller address  for this specific driver.

Medallions

The management of medallions allows the administrator to create a list of medallions owned by the facility, manage each of their characteristics, and finally to assign them a holder (patient). These medallions are used as portable and mobile devices to trigger nurse calls or runaways.



Just like for the nurse call devices management screen, the medallion administration screen displays a list of medallions on the left of the screen, and a detail area on the right presenting the characteristics of the medallion currently selected. An add button  at the top of the list is used to declare a new medallion, while deleting or editing an existing medallion is accessible via the contextual menu in the details area (vertical dots ). When editing the properties of a medallion, the administrator can enter the following properties:

- 1 **Name** identifying the medallion in a unique way throughout the software
- 2 **Identifier:** this is the unique address of the medallion, as known by the driver supporting it
- 3 **Device owner:** if the medallion is assigned, allows to select the user associated with the device (usually a patient or a resident)

- ④ **Driver instance** in charge of the communication with the device
- ⑤ **Nurse call type**: type of nurse call event to generate when the medallion is used to make a call
- ⑥ **Runaway events generation**: if he checks this box, the administrator indicates that the medallion will trigger runaways when passing in front of location beacons.
- ⑦ **System events generation**: This check box allows to indicate if one wishes that the loss of communication between the driver and the device causes the creation of a new system event. If this is the case, it is necessary to select ⑥ **the type of system event** to create from the list.

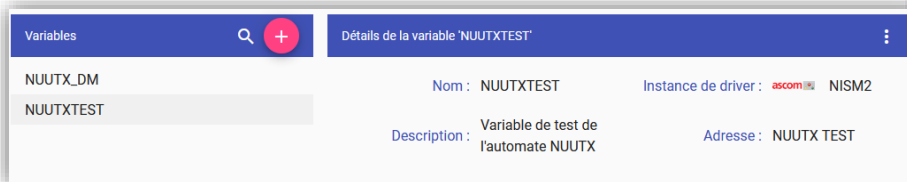
Event Generators

Event generators are Ofelia modules that create and evolve events when certain conditions of the execution context are met (see *Event Generators* on page 61). The administrator can define variables and write rules indicating the conditions for creating and evolving events according to the values of these variables. By mapping the value of the variables with external information received by drivers, it becomes possible to freely control the life cycle of events (alarms, for example) according to sensor or PLC data.

Variables

The variables, whose role is detailed in the chapter *Variables* on page 61, can be administered via a dedicated administration screen. This screen presents on the left a list of the variables already created in the system and, when one of them is selected, a detail area on the right. An add button at the top of the list  allows you to declare a new variable and a contextual menu in the details area (vertical dots ) allows you to delete or edit the variable currently displayed:

Editing a variable consists in filling in a name and a description and then associating it with a driver instance by indicating a unique address allowing this driver to browse or update it (depending on the driver's capabilities).



Alarms Generators

As indicated in chapter *Alarm generators* on page 61, these generators are made up of scripts indicating the state that the alarm should take according to the value of various input variables.

The alarm generators administration screen shows a list of generators on the left side of the screen, next to which there is an area showing the details of the selected generator. An add button  at the top of the list allows you to create a new generator, while a contextual menu (vertical dots ) allows you to delete the selected generator or edit its characteristics.

Editing an alarm generator consists in setting the following elements:

- ❶ **Unique Name** of the generator
- ❷ **Icon** used to represent the generator on the displays.
- ❸ **Zone**: Area where the generator is located. If the generator was added directly from the settings screen of this zone (see *Adding an alarm generator* on page 92), it can be moved.
- ❹ **Type of event** generated: type of alarm that will be created when the generator is triggered.
- ❺ **Parameters**: if selected event type requires any.

Once the general information of the generator is defined, the administrator can indicate for each possible alarm state a Boolean expression that should bring the alarm in this state. Each logical expression is a script that allows the combination of variable values (see *Variables* on page 61) and operators.

Those operators can be arithmetic

==	Egal
<	Strictly Lower
>	Strictly Greater
<=	Lower or Equal
>=	Greater or Equal
!=	Different

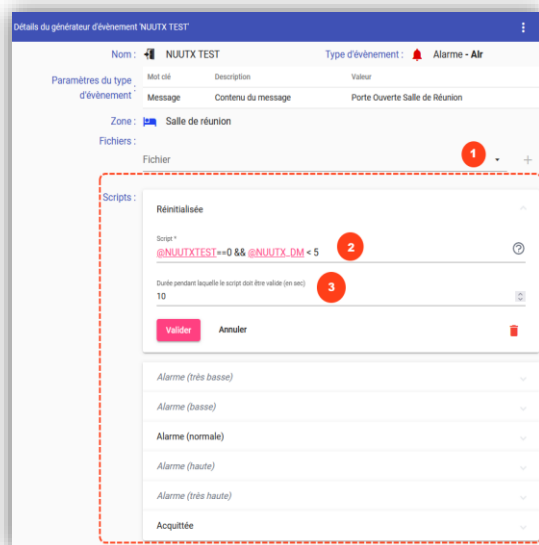
Or logical

&&	Logical AND
	Logical OR

A condition can be written for each alarm state:



The use of the @ character (at) allows to display a variable selection list in the entry zone.



1 **Files:** these will be automatically attached to each alarm instance created by this generator.

2 **Script:** The logical conditions expressed for each alarm state are analyzed in the following order: *Alarms Very high* => *High* => *Normal* => *Low* => *Very low*. As soon as one of them is verified, the others are no longer verified. Whether one of the previous alarm states has been checked or not, the script for the acknowledgement is systematically evaluated. (See *Technical Alarms* on page 18)

3 **Duration in seconds:** if this time is filled in, the logical condition must be checked during the indicated time before the state of the event is modified. This makes it possible to avoid toggling on unstable sensors, or to condition an alarm if a value passes a threshold for a sufficiently long time.

Mobilization Generators

A mobilization generator behaves like an alarm generator, with the difference that it generates mobilizations (notifications) instead of alarms and that it can generate several events simultaneously in the same zone, which is not possible with an alarm generator (see *Mobilization Generators* above).

The configuration screen is very similar to that of other generators, namely a list of generators on the left, an add button  to create a new one, and a detail area on the right allowing to display, delete, or modify the selected generator via a contextual menu (vertical dots ). The edition of a mobilization generator requires the following properties:

- 1 **Unique name** of the generator
- 2 **Icon** used to represent the generator on the displays
- 3 **Zone**: Area where the generator is located. This zone will be used as the current zone for the Mobilization event.
- 4 **Type of event** generated: type of mobilization that will be created when the generator is triggered.
- 5 **Parameters**: if the type of event chosen requires any. This can be in particular the *Message* parameter, which exists in the Mobilizations root type. The value of this message can use *@variables* which will be replaced during the creation of the event by their current value.

Once the general properties of the generator have been filled in, the administrator has to define the conditions that allow the creation of a mobilization in the only valid state: *In progress*. Unlike the alarm generators, the operation of the mobilization generator is simpler: if the logical condition expressed in the Script field is verified, a workflow is immediately executed. The configuration is as follows:

- 1 **Files**: these will be automatically added as attachments to each mobilization created by this generator.
- 2 Only the **current state** can be associated with a script and a validity period of the condition

- 3 **Script:** Logical condition which, if checked, triggers the workflow indicated below. The use of the @ character (at) allows to display a list of variable selection in the input field. The script can contain arithmetic operators to evaluate a variable value:

==	Egal
<	Strictly Lower
>	Strictly Greater
<=	Lower or Equal
>=	Greater or Equal
!=	Different

And logical operators between different comparisons:

&&	Logical AND
	Logical OR

- 4 **Duration in seconds:** if this optional duration is filled in, the logical condition must be checked during the indicated time before the state of the event is modified. This makes it possible to avoid toggling back and forth on unstable sensors, or to only start a mobilization if a value passes a threshold for a sufficiently long time.
- 5 **Workflow** to be triggered when the script condition is verified during the defined duration.
- 6 **Parameters** (if any) of the workflow if some of them have been specified as variables (these are parameters that would have been filled in by a triggering rule).

Mobilization Campaign Generators

A mobilization campaign generator aims at starting a campaign whenever conditions expressed in a script are verified. These conditions can involve the value of different variables, combined with arithmetic and logical operators (just like for Alarm generators described above).

The difference with a mobilization generator, which simply triggers a workflow, is that the campaign generator can start several mobilizations simultaneously. To this end, a mobilization campaign generator must embed in its configuration all the parameters and arguments necessary for all the workflows that it may have to trigger: in fact, once the triggering condition is met, no human intervention is necessary.

The administration of the campaign generators is done as for the other generators by a screen which presents on the left-hand side a list of generators with an add button  to create a new one, as well as a detail zone on the right allowing to display, delete or modify the selected generator via a contextual menu (vertical dots ). The edition of a mobilization campaign generator requires to fill in the following properties:

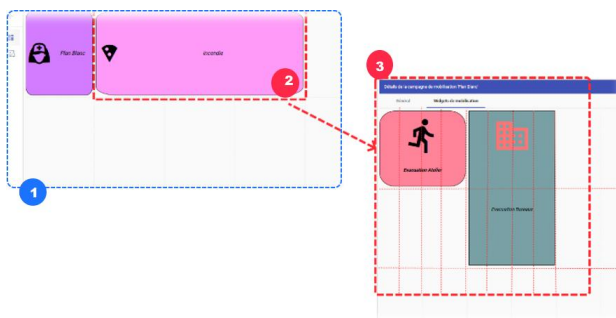
- ❶ **Name** of the generator
- ❷ **Icon** allowing to represent it in the displays (among a predefined list)
- ❸ **Zone** where the campaign is attached
- ❹ **Type of mobilization campaign event** to create, and possibly a list of event type parameters.
- ❺ **Mobilization campaign template**: Selected from the mobilization campaigns (See *Mobilization Campaigns* on page 59)
- ❻ For each widget of the mobilization campaign, **list of parameters** of the corresponding mobilization types, as well as parameters of the workflows of each widget. In particular, the basic mobilization type has a **Message** ❷ parameter that can be filled in either by a fixed text or by a text including variable values (using the @ character allows the list of available variables to appear).

Mobilization Campaigns

Campaigns offer a simple way for the administrator to provide operators with a dashboard, made up of different graphical buttons, allowing them to easily trigger a coherent set of mobilizations. The general philosophy of this functionality is detailed in the Mobilization Campaigns chapter (on page 59).

Campaigns Dashboard

The dashboard is a central element that acts as a "Meta Dashboard" of campaigns. Each campaign has its own graphical layout of buttons (widgets) to launch each individual campaign.

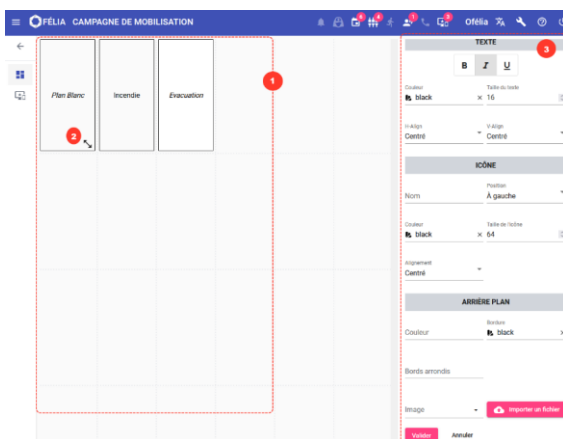


- 1 The campaign dashboard presents graphic buttons corresponding to each campaign
- 2 Pressing a button will open the corresponding campaign
- 3 This campaign will in turn have buttons (widgets) each triggering a mobilization

The administration screen of the campaign dashboard is presented in the form of a 6-column wide

grid on which are placed as many buttons as there are mobilization campaigns declared in the software (See *Campaigns dashboard* on page 60).

By default, these campaigns are each represented by a button with the name of the campaign in its center, and occupying a cell of the grid. When selecting one of these buttons, a contextual icon at the bottom right of each box allows you to resize it, and an editing pane slides in from the right of the screen. This gives access to the different editing functions of the campaign button.



- 1 Campaign buttons **display grid**.
- 2 **Resize handles**: by holding down the mouse button on this icon, the administrator can resize the button and make it occupy the desired number of cells.
- 3 Campaign button style **editing pane** (see below).

Once one of the campaign buttons is selected, the editing pane allows the administrator to define the graphic rendering of the button by editing the following text, icon and background properties:

Editing how the name of the campaign will be rendered as a button

- 1 **Bold** the name
- 2 *Italics* the name

TEXTE

1 **B** 2 *I* 3 U

Couleur 4 black 5 Taille du texte × 16

H-Align 6 Centré V-Align 7 Centré

ICÔNE

Nom 8 Position 9 À gauche

Couleur 10 black 11 Taille de l'icône × 16

Alignement 12 Centré

ARRIÈRE PLAN

Couleur 13 Bordure 14 black ×

Bords arrondis 15

Image 16 Importer un fichier

- 3 Underline the name
- 4 Change the **font color** of the name
- 5 Change the **size of the text**
- 6 **H-Align**: Horizontal Alignment (centered, left or right justified)
- 7 **V-Align**: Vertical Alignment (centered, top or bottom of the box)

Editing the (optional) icon displayed in the button

- 8 **Name**: Choice of the icon in a predefined list
- 9 **Position** in relation to the text: left, right, below, above
- 10 **Color** of the icon
- 11 **Size of the icon** in pixels (by default 16 x16). The icons have a square format.
- 12 **Alignment** in the slot: centered, top or bottom

Editing background

- 13 **Background Color** (white by default)
- 14 **Border color** (black by default, if no color provided the button has no visible border)
- 15 **Rounded edges**: in pixels or percentage, value of the curvature (up to 50%, which is an oval fitting the box)
- 16 **Picture**: Selection of an image file (JPEG, BMP, GIF, including animated GIF) that will be displayed in the box.

Note: it is not possible to import the same file several times, the files already imported are available in the Image drop-down list.

Campaign Dashboard

The campaign dashboard allows the administrator to create new campaign templates, and to edit the graphical dashboard of each of them by arranging the widgets giving access to the mobilizations that compose them.

The configuration is done in two steps: first the creation of the campaign itself, then the edition of the dashboard by adding the different mobilizations and their parameters while specifying their graphic representation. The administration screen itself is composed of a list of existing campaigns on the left, and a detail area on the right displaying the properties of the selected element in a *'General'* tab, a second tab *'Mobilization widgets'* allows to draw the campaign screen. An add button at the top of the list  gives access to the creation of a new template, while the modification of the campaign template selected in the list is done by means of a contextual menu (vertical dots ).



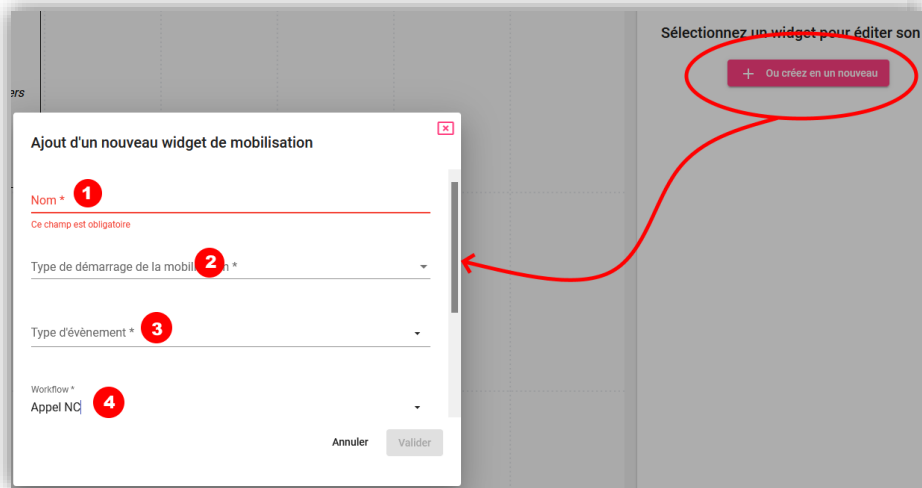
- 1 Campaigns list
- 2 General: Global settings for the campaign
- 3 Campaign Dashboard edition

The creation of a campaign template or the edition of its parameters requires to fill in the following information in the *'General'* tab:

- 1 **Unique Name** of the campaign template (this name will be used by all instances generated by this template).
- 2 **Unique Code** of the campaign template (code used by all instances from this template).
- 3 **Zone** the campaign event created by this template will be attached to (required field but can be attached to the facility root).
- 4 **Campaign Event Type** that will be used by the campaigns created by this template
- 5 **Event type parameters** (if the chosen type requires any).

The tab entitled *'Mobilization Widgets'* allows you to place the mobilization trigger buttons within the campaign, on a 6-column graphic grid. A button allows you to add a new element (aka *'widget'*) in the first empty box. This widget, representing a trigger button, can then be resized or deleted by clicking on the buttons representing a double arrow  or a trash can  respectively. A drag-and-drop function also allows you to grab a widget and move it to another cell of the grid. When selecting a widget, a customization pane slides in on the right side of the screen, giving access to graphic properties.

The creation of a new widget or the editing of an existing widget (using the contextual button ) allows you to define the general characteristics of the mobilization events that will be created when the button is pressed:



- 1 **Name** of the mobilization event
- 2 **Mobilization Start Mode**: Manual (when the button is pressed) or Automatic (as soon as the parent campaign starts)
- 3 **Type of mobilization event** that will be created when the button is pressed. If the event type has parameters, they appear here to be filled in
- 4 **Workflow**: selection of the workflow that will be executed when the mobilization event is created. If the workflow has parameters, the list is also displayed to allow them to be filled in.

The graphical customization pane is identical in all respects to the buttons in the campaign dashboard previously described:

TEXTE

1 **B** 2 *I* 3 U

Couleur 4 black Taille du texte 5 16

H-Align 6 Centré V-Align 7 Centré

ICÔNE

Nom 8 Position 9 À gauche

Couleur 10 black Taille de l'icône 11 16

Alignement 12 Centré

ARRIÈRE PLAN

Couleur 13 Bordure 14 black

Bords arrondis 15

Image 16 Importer un fichier

Editing the name of the mobilization that appears in the button

- 1 **Bold** the name
- 2 *Italics* the name
- 3 Underline the name
- 4 Change the **font color** of the name
- 5 Change the **size of the text**
- 6 **H-Align**: Horizontal Alignment (centered, left or right justified)
- 7 **V-Align**: Vertical Alignment (centered, top or bottom of box)

Editing the (optional) icon displayed on the button

- 8 **Name**: Choice of the icon in a predefined list
- 9 **Position** in relation to the text: left, right, above or below
- 10 **Color** of the icon
- 11 **Icon Size** in pixels (by default 16 x16). Icons have a square format.
- 12 **Alignment** inside the slot: centered, top or bottom

Editing Button Background

- 13 **Background Color** (white by default)
- 14 **Border color** (black by default, if no color provided the button has no visible border)
- 15 **Rounded edges**: in pixels or percentage, value of the curvature (up to 50%, which is an oval fitting the box)
- 16 **Picture**: Selection of an image file (JPEG, BMP, GIF, including animated GIF) that will be displayed in the box.

Note: it is not possible to import the same file several times, the files already imported are available in the Image drop-down list.

System Administration

Ofelia® is based on a technical environment (*Microsoft Windows®*, *Microsoft SQL Server®*, etc.) whose administration goes beyond the recommendations given in this document. Configuration prerequisites are presented in the installation guide.

However, the software's graphical interface makes it possible to carry out a number of system administration operations and to automate common maintenance tasks: loading and updating the license, purging data, backing up database, etc.

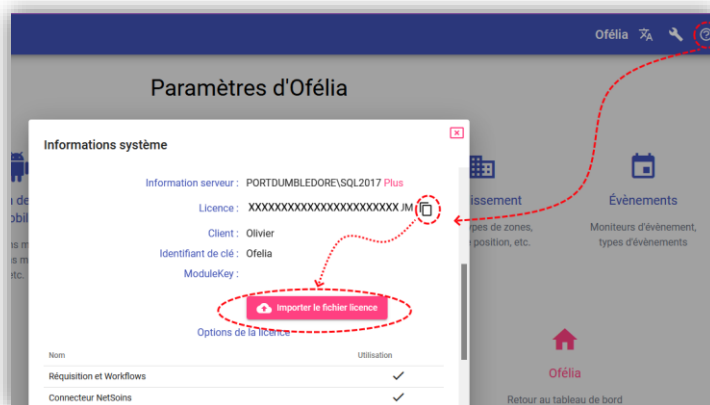
License management

The software license is activated by means of a file that must be uploaded to the server. This file is obtained by providing the sales department with a unique environment code, linked to the hardware installation of Ofelia on site. Any new installation on new hardware renders the license null and void and must be regenerated.

If there is no valid license file, the application goes into evaluation mode for a limited time before shutting down completely. The evaluation mode can be identified by a colored banner at the top of the screen when the user logs in.

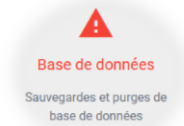


The access to the download of a license file is done through the 'About' button located on the upper toolbar (See *Product Information* on page 178). A shortcut (⌘) allows you to copy the environment code to the clipboard and send it to the editor. In return, the editor provides a file (.lic extension) that can then be imported into the program using the button depicted below:



Data Management

A number of database administration functions are available in a dedicated area of the software administration. The 'Databases' menu gives access to the database backup and restore features, import and export of the configuration as well as the management of the retention policy. To warn of the danger of the manipulations carried out in this zone, the screen is displayed in red.

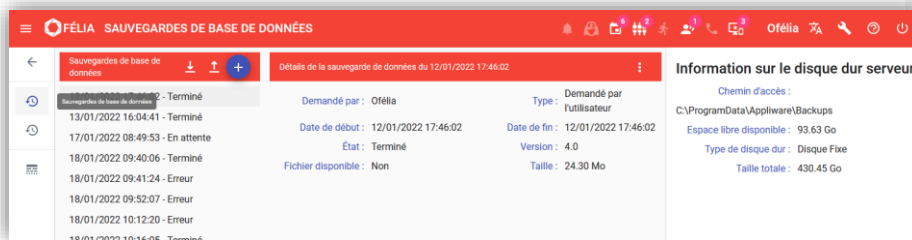


Base de données
Sauvegardes et purges de
base de données

Backups

In the data administration screen, the backup and restore features are accessed via the dedicated button in the side toolbar on the left of the screen.

The screen is presented as a list of backups, next to which is a detail area showing the characteristics of the selected backup. A more general information pane is located on the right-most part of the screen: it displays the location where the backups will be stored and indicates the total occupied space and the disk space still available in case the backup location is a local disk of the Ofelia[®] server (if it is a network drive, the disk space information is not available). At the top of the list of backups, an add button (+) allows, after confirmation, to launch an immediate backup operation of the database.



The detail area of the history of previously performed backups shows the following information for each item:



- 1 **Requested by:** User who requested the immediate backup or user who scheduled the backup
- 2 **Start date:** date and time the backup was started (manual or scheduled)
- 3 **State:** In error, In progress or Completed.

- 4 **Available File:** indicates whether the corresponding file is still present on the disk. If the file has been deleted, only the metadata is present and thus it will not be possible to restore the backup.
- 5 **Backup Type:** Manual or Automatic.
- 6 **End Date:** Date on which the backup operation ended (allows you to know how long the backup lasted).
- 7 **Version:** Version of the database schema being backed up.
- 8 **Size:** Size of the backup file (.Bak) on the disk.
- 9 **Context Menu** for the selected backup, allowing to delete the backup and its metadata. If the corresponding file is still present on the disk, this menu also offers an option to **restore** the backup.

The restoration of a backup starts by displaying a confirmation request: its acceptance causes the immediate disconnection of users until the end of the data restoration operation, and the replacement of active data by those present on the backup set.

Notes:

- No backup of the current dataset is automatically performed before restoring an archive. All production data at the time of restoration is therefore lost.
- It is possible to perform a backup while the software is in production, but it is not recommended to do so during a period of high activity because the high demand on the server and the disk generated by this operation may lead to random errors on the connections in progress.

Target Directory

If nothing is specified in the configuration file, the backups are placed in the [ProgramData\Appliware\Backups] directory of the server where Ofelia[®] is installed. In the case of a deployment where *Microsoft SQL Server*[®] is located on another server than the application server, it is necessary to indicate the location of the backups in the configuration file as a network share and to ensure that both the Ofelia[®] and SQL Server[®] service accounts have proper write access to both the folder and the share.

Compatibility

When restoring a backup, its version is compared to that of the software being run: if the database is older than the software version, it is first updated and the data is migrated to the target structure at the next restart. If, on the other hand, the database is more recent, the application refuses to start until it has been updated to support this version of the database, thus preventing any risk of data corruption that might occur due to an unsuitable or obsolete program.

Scheduled Backups

In addition to the manually initiated backups, the administrator can also schedule backups with a predefined periodicity. This allows them to be performed outside of the software's busy hours (see remarks above), at night for example, and to be coordinated with an external backup of the disk on which the backup files are stored (to a tape drive, a SAN, or a cloud).

All the backups performed follow a 'Full Backup' policy, for the implementation of a *Differential Backup* policy, the administrator is invited to perform the configuration directly within the *Microsoft SQL Server®* engine.

The screen for programmed backups, which can be accessed via the corresponding button on the left sidebar, also consists of a list of programs (on the left) and a detail area showing information about the program currently selected. At the top of the list of programs, an add button (+) allows you to create a new schedule, while the modification of an existing schedule is accessible via the contextual menu in the details area (vertical dots ⋮).



The creation or edition of an existing schedule is done by filling in the following properties:

- 1 **Name** of the schedule.
- 2 **Number of old backups to keep**: beyond this number the first one will be deleted, and so on in a circular fashion (FIFO).
- 3 **Time of the day** when the backup is to be started.
- 4 **Type of periodicity**, details appear in lower area (5).

The proposed periodicity is daily, weekly, monthly or yearly, the most commonly used being daily or weekly backups (if some days of the week have no significant activity to be backed up). The different modes are given below:

Every Day

TOUS LES JOURS

Tous les 1 jours

À partir du 15/2/2022

Toujours

- 1 Backup **interval** (e.g., every 2 days, every 5 days, etc.).
- 2 **Start Date** of the schedule.
- 3 **Duration**: *Always* (without end date), *Until a certain date* to be specified, or for a given *number of occurrences*.

Every Week

TOUTES LES SEMAINES

Toutes les 1 semaines

Lundi Mardi Mercredi Jeudi Vendredi

Samedi Dimanche

À partir du 15/2/2022

Jusqu'à une certaine 22/2/2022

- 1 Backup **interval** (e.g., every 2 weeks).
- 2 **Days of the week** when the backup takes place (multiple selection).
- 3 **Duration**: *Always* (without end date), *Until a certain date* to be specified, or for a given *number of occurrences*.

Every Month

TOUS LES MOIS

Tous les 1 mois

Indiquez le jour du mois: Le premier

À partir du 15/2/2022

Pour une série d'événements 5

- 1 Backup **interval** (e.g., every 6 months).
- 2 **Day of the month** by giving the number of the day or the occurrence of the day of the week (for example the Second Wednesday of the month)
- 3 **Start Date** of the schedule.
- 4 **Duration**: *Always* (without end date), *Until a certain date* to be specified, or for a given *number of occurrences*.

Every Year

TOUS LES ANS

Tous les 1 ans

Indiquez le jour du mois: Janvier

Le premier Lundi De Janvier

À partir du 15/2/2022

Pour une série d'événements 5

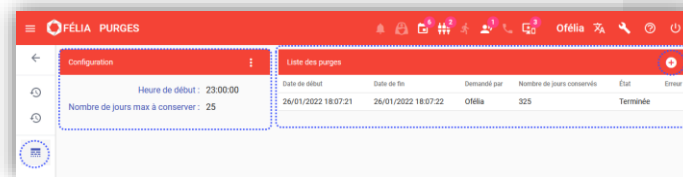
- 1 Backup **interval** (e.g., every 2 years).
- 2 **Day of the year** by giving the number of the day and the month or the occurrence of the day of the week and the month (for example the Second Wednesday of January)
- 3 **Start Date** of the schedule.
- 4 **Duration**: *Always* (without end date), *Until a certain date* to be specified, or for a given *number of occurrences*.

Retention Policy

The implementation of a traceability solution inevitably leads to an inexorable increase in the volume of data stored. To ensure that the system maintains optimal performance and to avoid keeping unnecessary obsolete data, the administrator can choose to implement a data retention policy by regularly purging expired data. These purges can be launched manually, or be part of the general retention policy: a scheduled purge can be launched on a daily basis to ensure that outdated data is deleted.

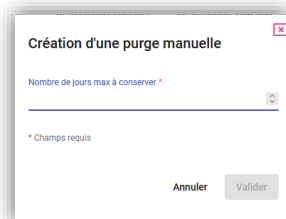
The purge management functionality is accessed via a button on the left sidebar (🗑️).

The screen is composed, on its left-hand side, of an area allowing to manage the time of the scheduled purge and the number of days to be kept within the retention policy of the site. On the right side, a history of the purges already performed is displayed along with their status (date, applicant, number of days retained, status, and possibly the error message). An add button at the top of this list allows you to create an immediate purge.



Manual Purge

Pressing the immediate purge button (+) opens a window for entering the **number of days to be kept**: all the data related to events older than the number of days specified will then be definitively deleted.

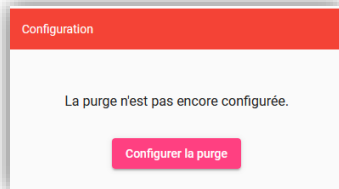


Configuration data is not affected. The purge operation, although accessible during production hours, causes a significant load on the database (depending on the extent of its date range): it is therefore far preferable to launch it during a period of lower activity in order not to impede other processing.

Scheduled Purge

To ensure that old data is regularly deleted from the database, the administrator configures automatic purging: this involves defining how many days of event history are kept in the system at most. Even though the fact that the scheduled purge operation is carried out daily means that its impact is low (maximum of one day's data deletion), it is still preferable to carry it out outside of the software's busy hours. Moreover, its implementation must be synchronized with any other administrative operations on the server (backups, transfers, etc.): the administrator can therefore set the time at which the daily purge will take place.

By default, no purge is configured when the user arrives on the administration screen, a button allows to set it up:



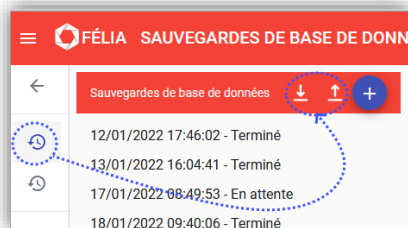
When an automatic purge is configured, a contextual menu (vertical dots ) allows you to delete it or modify its characteristics, i.e., to define the number of days to keep and the time of its execution.



Configuration Backup

The backups and restores presented above are an automation of the backup processes internal to the *Microsoft SQL Server*® database engine, and concern both the operational data (current events, history, etc.) as well as the software configuration. Configuration backup is a different mechanism that only targets the settings made by the administrator (users, zones, media, etc.). The scenarios covered by this functionality have been presented previously (see *Configuration Import and Export* on page 70).

The access to the export and import functions of the software configuration is done through the same screen that allows to manage the backups: at the top of the list of backups, there is **(a)** an icon representing a downward arrow that allows to download an export file of the configuration and **(b)** an upward arrow that allows to upload an existing file into the software.



When the administrator uses the export button () the browser immediately downloads the generated file. Depending on the browser used, the administrator can choose where to store the file or retrieve it from the *Downloads* folder on his workstation.

When the administrator uses the import button () , he must first confirm his action before selecting a file: the import function is indeed destructive and cannot be cancelled.

Confirmation d'action

ATTENTION, toutes vos données actuelles vont être effacées après le chargement du script de configuration. Êtes-vous sûr de vouloir continuer ?

Annuler**Valider**

If the user confirms the import, the system downloads the file, analyzes its contents to validate that the version of the imported configuration is compatible with that of the application, then the users are disconnected from the system, all the data is deleted and the new configuration replaces the current one. At the end of the import, users can log back in to the system and start again with a completely new database in terms of history.

Operator's Guide

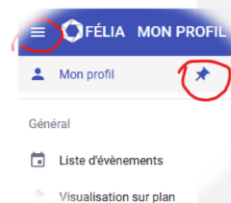
Navigation

When a user logs in, unless a specific landing screen has been specified in the user's profile (See *User Profiles* on page 113), the *Events In Progress* screen (see on page 180) is immediately displayed.

The upper navigation bar of the screen is common to all screens and includes the following status and navigation elements:

Événements	Événements cachés	Événements des 8 dernières heures
##	31/01/2022 09:35:09	14mins Driver Instance défaut système Actif S_9089 #PORTDUMBLEDORE Ackermann (Système) Driver Instance défaut système Etablissement
##	31/01/2022 09:35:02	15mins Driver Instance défaut système Actif S_9088 #PORTDUMBLEDORE NISM2 (Système) Driver Instance défaut système Etablissement
##	31/01/2022 09:35:00	15mins Driver Instance défaut système Actif S_9087 #PORTDUMBLEDORE IMS3 (Système) Driver Instance défaut système Etablissement
##	31/01/2022 09:34:59	15mins Driver Instance défaut système Actif S_9086 #PORTDUMBLEDORE Email (Système) Driver Instance défaut système Etablissement
##	28/01/2022 15:38:35	2j 18h 11mins Missions parti M_9081 : Missions CH 08 Ofélia Prév
##	25/01/2022 16:45:25	5j 17h 4mins Campagne de Mobilisation En cours MB_9048 Campagne de Mobilisation Etablissement Ofélia
##	25/01/2022 16:43:47	5j 17h 6mins Campagne de Mobilisation En cours MB_9046 Campagne de Mobilisation Etablissement Ofélia te
##	17/01/2022 10:15:43	13j 23h 34mins Missions parti M_2788 : Missions Etablissement 449 449 Ofélia

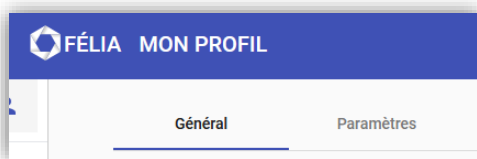
- Event indicators:** an icon for each type of event with an information bubble giving the number of elements in progress for each. Clicking on the icon gives access a dashboard displaying the current events (see *Events In Progress* on page 180) already filtered by the type selected
- Name of the user** connected to the application, access to user profile edition (See *User Profiles* on page 113)
- Change the **language** of the application
- Access to the application **configuration area** (See *Administrator Guide* on page 80)
- Access to **product information**: version, details of the license and active options, etc. (See *Product Information* on page 178)
- Logout** current user
- Expanded menu** icon: displays an expanded version of the menu in a sliding pane with labels next to each icon. A pin maintains this pane open.



Depending on the licenses acquired and connected user permissions, all the elements presented in this guide will not necessarily be visible in the graphic interface.

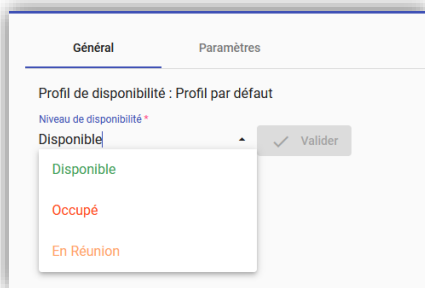
User Profile

By clicking on the name that appears in the upper right-hand corner of the top bar, the user has access to the profile settings window. This window is split in two tabs (*General* and *Settings*) that allow the user to set their level of availability as well as various display preferences in the software.



Availability

The '*General*' tab gives access to the user's availability level, it allows him to change instantly the availability status that will be interpreted by the requisition engine, indicating for example that he is '*available*', or '*in a meeting*'. The availability levels presented are those of the availability profile to which the user is attached. The profile itself is displayed but can only be changed by the administrator (See *Availability Profiles* on page 111):



More information on how availability profiles and levels work (and how they affect the way the engine requisitions personnel) can be found in the chapter *Availability* on page 38 and *Availability Levels* on page 111.

Display Settings

The display parameters tab allows the user to define how he wants to configure the software's graphical interface: this configuration is maintained each time he visits the software, regardless of the changes he may make during his browsing (for example, on the pagination). For each option there is a default built-in configuration of the software to which he can return by a simple toggle, allowing him to invalidate his customizations.

1 Landing page: The first page displayed after logging in, among those accessible from the sidebar (see Navigation on page 175).

2 Open navigation menu: Indicates whether the sidebar should be opened or closed with only icons (see Navigation on page 175).

3 Deletes customizations and returns to the **default configuration**: i.e., current events as a home page and reduced sidebar.

4 Number of hours to consider for the "recently completed" events tab of the current events view (See *Events in progress* on page 180). Default is 8 hours.

5 Number of items displayed on each page of the lists. If nothing is specified, the default value is 20 items per page.

6 Choice of the number of items proposed in the selection box displayed at the bottom of each paginated list. This has to be entered in increasing numbers separated by commas (for example 20,50,100). If nothing is specified, the default value is 5,10,15,20,25,50,75, 100.

7 Choose the filter for displaying the view by department (see *Department / Unit View* on page 189). This filter will be applied each time the user opens this view, regardless of any ad hoc changes he may have made.

8 Filter default value: By default, built-in

settings show only areas with event generators and belonging to a service and exclude hidden events

9 For event lists that allow **export to CSV or PDF format**, the icon  allows to select which columns are exported and in which order.

10 **Enable sound notifications:** if this box is unchecked, no sound will be played in the interface (see *Sound Profiles* on page 115).

11 If a user logs in and many unnotified events are pending, **limit the number of events that will result in a sound** warning to the specified amount.

Product Information

The 'About' icon (🔗) of the main banner opens a window summarizing the software version and deployment information. Most of this information is essentially necessary for technical support in case of diagnosis, others allow to know the state of consumption of dynamic licenses. Finally, a button allows an authorized administrator to update the license by importing a new file.

1 Software Version (refer to *Installation* on page 68 for the meaning of the numbering).

System information

1 Software Version: 1.2 Rev 3

2 Functional areas activated in the license:

- Alarm
- LWP Alarm
- Mission
- Mobilization
- Mobilization Campaign
- Nurse Call

3 Database Name: Database 'OfeliaE' : Version '1.2.3.0'

4 Database Schema Version: Database server info: xxxxxxxx\SQL2017 More

5 Name of the SQL Server instance (the link gives access to additional information: database engine version, etc.)

6 Unique environment code to be communicated to technical or commercial support when generating a new license. The button at the end of the line copies this number to the clipboard.

7 Name of the customer registered in the license file.

8 Customer key as found in the license file.

9 ModuleKey: If a hardware key is configured, its information is presented.

10 Button to import a license file: it allows to select a file (.lic extension) and to update the license. Users will be disconnected and will have to wait for the license update to be completed.

11 Options activated in the license. For the options with a volume limitation, the quantity acquired is displayed.

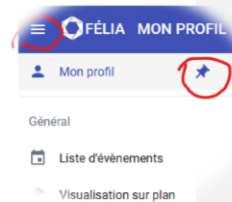
12 Real-time Consumption of licenses by volume, allowing to control the use and the adequacy of the chosen license with the real need.

Name	Usage
Requisition and Workflows	✓
NetSoins Connector	✓
Variables	100
Statistics	✓
ESPA Out	✓
Voice channels	10
Number of user medias	3
External locations providers	✓

Name	Usage
Input driver instances	5 / 350
Technical alarms	2 / 100
LWP Alarms	3 / 50
Mobilizations	1 / 50
Nurse calls	21 / 350
Number of manageable medias	0 / 100
Callable users	3 / 500

Navigation

The sidebar on the left of the screen gives access to the different operating screens of the software. Depending on the license acquired and the consultation rights of the connected user, not all the icons presented here will be present. The bar can also be deployed as a side panel to display an explanatory text under each icon, this mode can be "pinned" to keep it open. The behavior of the bar (collapsed or expanded) is one of the elements that can be customized for each user in their profile (see *Display Settings* on page 176).



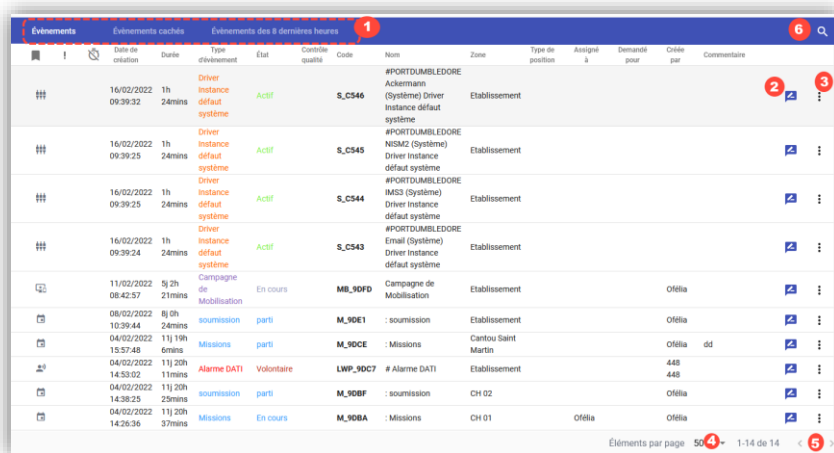
The icons in the sidebar provide access to the following features:

	Access to the profile of the connected user (See <i>User Profile</i> on page 176)
	List of events in progress (see <i>Events in progress</i> on page 180)
	Floor map view
	Department View
	Temporal Alarms
	Technical Alarms
	Mobilization Campaigns
	Missions in progress
	Scheduled missions
	Patients Management
	Customers Management
	Staff members Management
	Event History
	Statistics

Events in progress

Dashboard

By selecting the corresponding option from the side menu () or by clicking on one of the icons in the notification area located on the top bar (see *Navigation* on page 179), the current events screen is presented to the user. This screen displays a paged and filterable list of unfinished events, as well as a list of events that have recently finished (the number of hours can be set in the user's profile, see *Display Settings* on page 176). This list is restricted to the events that the current user has the right to consult.



Événements	Événements cachés	Événements des 8 dernières heures	
Date de création	Durée	Type d'événement	Statut
16/02/2022 09:39:32	1h 24mins	Driver Instance default système	Actif
16/02/2022 09:39:25	1h 24mins	Driver Instance default système	Actif
16/02/2022 09:39:25	1h 24mins	Driver Instance default système	Actif
16/02/2022 09:39:24	1h 24mins	Driver Instance default système	Actif
11/02/2022 08:42:57	5h 21mins	Campagne de Mobilisation	En cours
08/02/2022 10:39:44	8h 24mins	soumission	parti
04/02/2022 15:57:48	11h 19h 6mins	Missions	parti
04/02/2022 14:53:02	11h 20h 11mins	Alarme DATI	Volontaire
04/02/2022 14:38:25	11h 20h 25mins	soumission	parti
04/02/2022 14:26:36	11h 20h 37mins	Missions	En cours

1 Tabs used to display either:

- The list of **unfinished events** (by default). This list does **not** contain the events whose 'hidden' attribute is active.
- The list of **hidden events**. These are events in progress that the user has chosen to remove from the main screen or whose event type indicates that they should be hidden by default as soon as they are created.
- The list of **events completed in the last few hours**. This tab provides quick access to the list of events that have recently ended without having to browse the history. The number of hours to consider a completed event as 'recent' is a configurable parameter in the user's profile, by default only the events of the last 8 hours are present (see *Display Settings* on page 176).

2 Access to the **event handbook** to browse time-stamped comments or append new ones.

3 Access to **quick actions** on the event (see *Quick Actions* on page 183).

4 Choice of the **number of items displayed on each page** (see *Display Settings* on page 176)

5 Access to the **next or previous page** of the list, based on the column used for sorting.

6 Access to the **records filter** (see *Filters* on page 182)

Columns

The rows of the dashboard have 16 columns, and are terminated by a contextual menu (vertical dots ) giving access to quick actions on the event concerned (see *Quick Actions* on page 183).



By clicking on the header of a column, the user can reorder the list by alternately sorting upwards or downwards by the content of that column. A pictogram in the table header indicates the column used for sorting as well as its direction

The content of each column of the general tab and the hidden events tab is as follows:

Column	Description
	Nature of the event represented by its icon:  System Event  Nurse Call  Runaway  LWP Alarm  Alarm  Mission  Mobilization  Campaign
	Priority of the event represented by the icon and the color defined by the administrator (See <i>Event priorities</i> on page 140)
	For missions only, indicates if an Out-of-time monitor has been triggered (see <i>Out-Of-Time Monitors</i> on page 47) by the presence of an icon in the column
Creation Date	Date and time of creation or generation of the event
Duration	Time elapsed since the event was created
Event Type	Name of the type of event, written in the color defined for the type
State	Name of the current state of the event, written with the color defined for the state (See <i>Event state definitions</i> on page 140)
Quality-of-Service	If a default quality control for the event type applies (see <i>Quality-of-Service Monitors</i> on page 135), its level is indicated here, along with its icon, and its name (in the color of the level))  Trop long
Code	Unique code generated to identify the event
Name	Name generated for the event according to the formatting rule for its type (See <i>Common Properties</i> on page 129)

Zone	Current zone of the event, i.e., its zone of attachment at creation or the zone of presence in real time in case of a runaway
Position Type	Position inside the aforementioned zone
Assigned To	Requisitioned staff, as a comma-separated list of first and last names
Requested For	Requestor of the event (patient for a nurse call, staff member for an assignment, etc.)
Created By	Creator of the event in case of a manual creation
Comment	Comment entered at creation (for a mission)

The tab showing recently completed events shows the exact same columns except that:

- the state column does not show the *current* state but rather the *initial* state when the event was created,
- the quick actions button is missing (because you cannot act on a completed event)

Filters

By activating the magnifying glass button (🔍) at the top right of the *Events in Progress* or *Hidden Events* tabs dashboard, a filter pane slides down from the top of the list to replace the title bar. A button at the top right of this filter (✕) closes it and returns to the standard view. The event filtering options are as follows:

The screenshot shows a filter pane for 'Événements' (Events). It contains several search and selection fields, each numbered with a red circle:

- 1**: Code (text input)
- 2**: Du (start date)
- 3**: Au (end date)
- 4**: Natures d'événement (multiple selection list)
- 5**: Types d'événement (multiple selection list)
- 6**: État (multiple selection list)
- 7**: Contrôle qualité (multiple selection list)
- 8**: Niveaux (multiple selection list, with a note: 'Choisissez le contrôle qualité avant de choisir ses niveaux')
- 9**: Priorités d'événement (multiple selection list)
- 10**: Zones (multiple selection list)
- 11**: Services (multiple selection list)

- Exact code** of the event to select. This option can only return one event, wildcards are not allowed.
- From**: start date of the event creation range.
- To**: end date of the event creation range.
- Event Natures**: multiple selection list. Only events of these natures will be returned.
- Event Types**: multiple selection list. The hierarchy applies: if an event type is selected, the events of this type and all the descendants of this type will be returned.
- States**: multiple selection list, among all the states of the event natures possibly selected in **4**.
- Quality-Of-Service**: return only the events which have satisfied the QoS monitor selected.

- 8 **Levels:** multi-select list. If a QoS monitor is selected in 7, refine to return only the events that have reached one of the indicated levels.
- 9 **Event Priorities:** multi-select list grouped by priority profile (See *Event priorities* on page 140). Returns only those events whose current priority is one of those selected.
- 10 **Zones:** multi-select list. The hierarchy applies: if a zone is selected, the events of this zone and of all the descendants of this zone will be returned.
- 11 **Departments/Units** multiple selection list. Only the events taking place in a zone of these departments are returned.

The filter is not present in the recently completed events tab: to search in this tab, the user can go to the event history dashboard (see *Event history* on page 208).

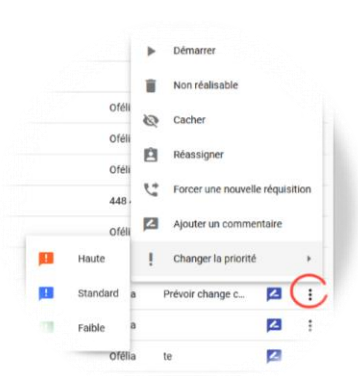
If the screen has been opened via the notification icons in the top bar of the program, the filter on the nature of the events is automatically set when the list is displayed.

Quick Actions

Depending on the context of each event displayed, being either the current state of the event itself or the permissions of the user browsing it, a certain number of actions are possible, such as:

- Change its priority, according to the priority group associated with the type of event (See *Event priorities* on page 140)
- Change its state, for example start a mission, acknowledge an alarm, etc.
- Add a comment (handrail) or an attachment
- Manually reassign a list of employees
- Force a new requisition
- Toggle the state of the hidden event indicator, allowing it to appear alternately in the current events tab or in the hidden events tab.

These actions are grouped together at the end of each event line in the dashboard inside a contextual menu (vertical dots ) , thus avoiding the operator having to open the event details to access them. The user knows instantly what he can do on each event, since only the actions that are compatible with both the current state of the event and its clearance level are displayed in this menu. The list of quick actions is not available on completed events.

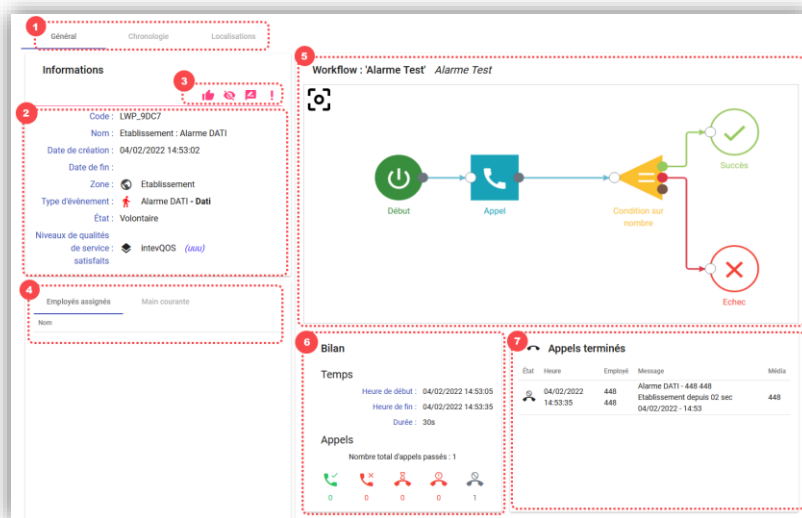


Event Details

Clicking on one of the events in the dashboard list opens a modal window. This window contains all the detailed information regarding the course of an event, split into three tabs: a **general** view, a **chronological** view, and a **geographical** view.

General View

The general tab view is divided into several areas:



- 1** **Navigation Tabs** between the general view, the chronology and the location of the event.
- 2** **Detail Information:** this area groups the event information already found in the columns of the dashboard list.
- 3** **Quick Actions:** these buttons correspond to the quick actions located at the end of each event row in the dashboard (see *Quick Actions* on page 183).
- 4** **List of employees** already assigned to the event, and in a second tab **time-stamped comments** of the event (in the form of a handbook).
- 5** Graphic display of the state of the **last requisition workflow**. If the workflow is not completed, its display is updated in real time to reflect the progress of the process: nodes and connections that have not been visited are grayed out and get colored as the process progresses. The user can move or zoom the workflow using the mouse and the scroll wheel, and a centering button (📐) brings the workflow back into the display dimensions.

Note: Selecting a workflow activity changes the display of the report (6) and call (7) fields so that they reflect only the contents of the selected node and not the entire workflow. This gives the user access to the precise details of each individual step of the workflow.



- 6 **Overall Summary** of the selected workflow or node in terms of execution time, parameters and number of calls made. Various icons illustrate the number of calls Accepted (✓), Rejected (✗), Expired (⌚), Failed (⚡), or Cancelled (⏸).

- 7 **Calls Details:** This list shows the status of each call made or rejected, specifying the reason for the employee's selection (e.g., because of his or her position, location, etc.) or omission (e.g., absent, incompatible availability, etc.). By clicking on a node, additional details of the various attempts and status changes are presented.

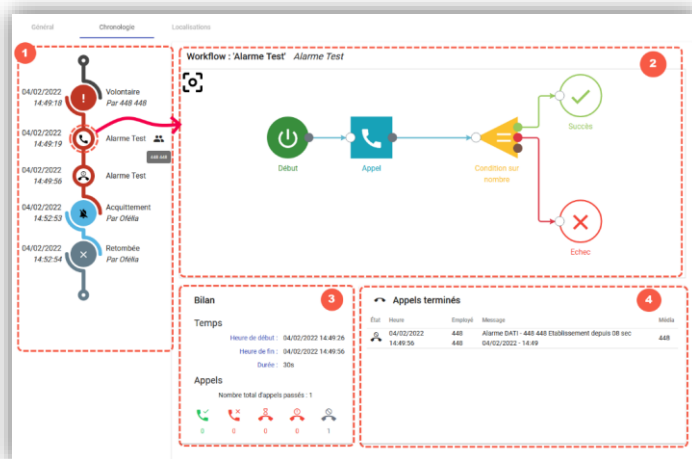
Appels terminés du nœud 'Appel'

État	Heure	Employé	Message	Média
Appel en cours	04/02/2022 14:53:35	448	Alarme DATI - 448 448	448
Appel en cours	04/02/2022 14:53:35	448	Etablissement depuis 02 sec	448
Appel en cours	04/02/2022 14:53:35	448	04/02/2022 - 14:53	448
Appel en cours	04/02/2022 14:53:35	448	Appel en attente d'acquittement - (300)	448
Appel en cours	04/02/2022 14:53:35	448	Annulé	448

Raison de l'appel : Créateur de l'événement

Chronological View

The view on the tab entitled *Timeline* gives access to the representation of all the state transitions that the event has undergone since its creation: there are automatic or manual state changes, priority modifications, monitor expirations, etc. For each transition, the requisition workflow that has been executed and the requisitioned staff are shown. The screen itself is divided into several areas:

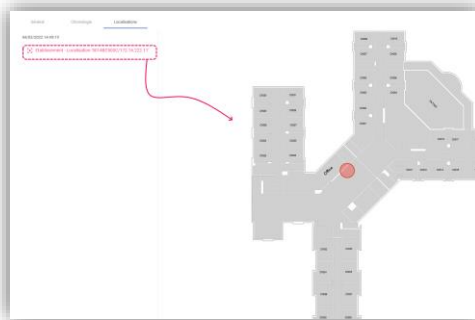


- 1 **Chronology** of the event's life on a vertical timeline. Each state change is represented by a large icon, while the beginning and end of a workflow or intermediate priority changes are represented by a smaller icon. To the left of each icon is a precise timestamp of the moment of transition, while to the right is a label describing the evolution of the event and possibly the user responsible for this transition. By selecting a step, the user can display -if applicable- the workflow triggered during the transition (in the frame 2). Finally, next to the workflow's start icon, a second icon shows the requisitioned staff in a tooltip.
- 2 Graphic display of the state of the **last requisition workflow**, identical in all respects to the one found in the general view tab (see previous chapter), or of the workflow currently selected in the timeline.
- 3 **Global overview** of the selected workflow or node and 4 **details of the calls**: same as General view (see previous chapter above).



Geographical View

This last tab of the event details shows a time-stamped list of the positions occupied by the event over time. For most types of events this list contains only one entry, but it can be more extensive for an event whose position evolves, such as a runaway for example: the successive movements of the fugitive will then be shown in the selection list. When a location is selected in this list, the corresponding map is displayed on the right-hand side of the screen:



By default, the tab opens on the last known location of the event (which simply means the location of the event itself for "static" natures).

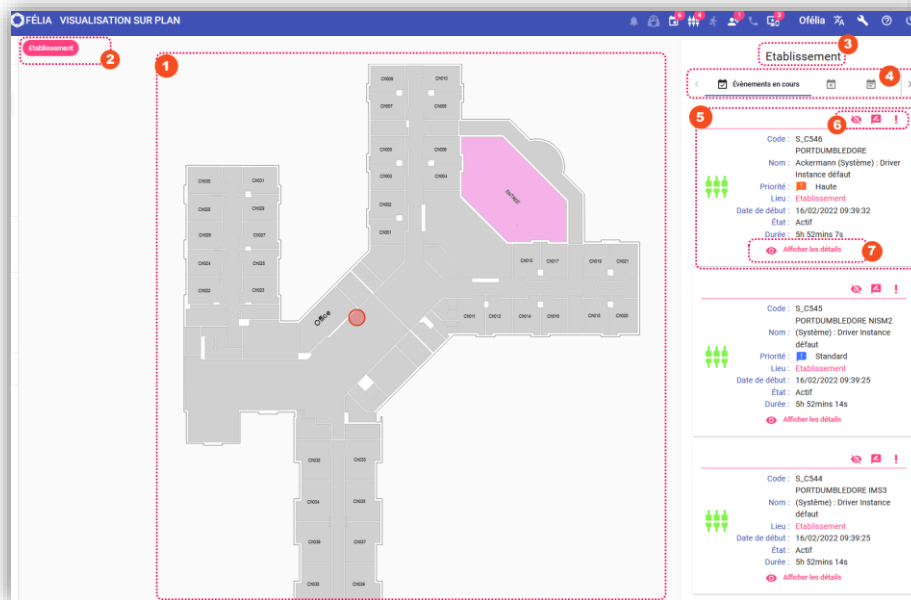
If the location is that of an area of the facility, the plan of the level directly above is displayed (see *Floor maps managements* on page 93). If the position is a GPS position captured by a beacon, a cartographic representation is shown.



Floor map view

By selecting the corresponding option in the side menu (), the software displays a screen locating events on the map. This screen allows the events in progress (i.e., not yet completed) to be displayed by positioning them on the plan of the facility. For this display to work, the administrator must have previously configured the facility's maps at all levels of the zone hierarchy (see *Definition of zones on the map* on page 94). This list is restricted to only those events which the current user has the right to consult.

The screen lets the user navigate through the different levels of the hierarchy of areas by browsing the events corresponding to the map displayed, and also, starting from an event located in a deep level of the hierarchy, to locate it immediately on the most relevant plan. The screen itself is made up of several zones:



- 1 **Graphic display** of the plan, highlighting the events in progress, in the color of their type or status according to the settings made by the administrator (see *Event state definitions* on page 140). By clicking on the highlighted area, the display moves to the plan level closest to the selected event.
- 2 **Bread crumb** indicating the location of the plan displayed in the zone hierarchy, and allowing to navigate to higher levels by simply clicking each sub-item.
- 3 **Name of the zone** currently displayed: all the events displayed in the list below take place in this zone or in one of its descendants.
- 4 **List Filter Menu**: current events , hidden events , recently completed events  (depending on the user profile settings, see *Display Settings* on page 176), event generators present in the zone , employees present in the zone .

Displaying events

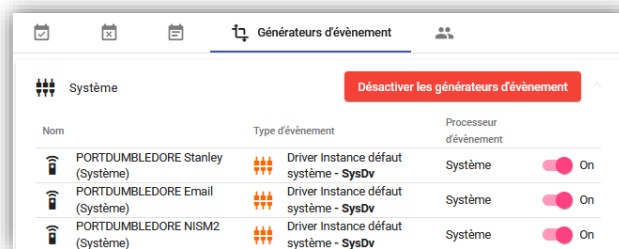
If the filter menu is set to current events, hidden events, or recently completed events, the list below shows the following for each event in the area currently displayed:

- 5 **Event details:** summary identical to the one found in the Information area of the details window (See *General View* on page 184)
- 6 **List of Quick Actions** that can be performed on the event (see *Quick Actions* on page 183)
- 7 Link to open the **event details window** (see *Event Details* on page 184)

If several events are located in the same display area, the one with the highest status priority determines the display color. This case of conflict is particularly frequent when the map is focused on a high level of the zone hierarchy (a site plan with several buildings for example): in this case, if the plan includes a polygon to represent a building, among all the events in progress in this building (alarms, but also missions, escapes, etc.) only one can give its color to the area representing the building. The choice of an appropriate prioritization of the states of each type of event is therefore essential for the readability of the whole (see *Event state definitions* on page 140).

Displaying event generators

If the filter menu is set to *event generators*, then the list below shows all event generators of the zone or its descendants (for instance, all nurse call generators of the rooms if the selection is a corridor or a floor):

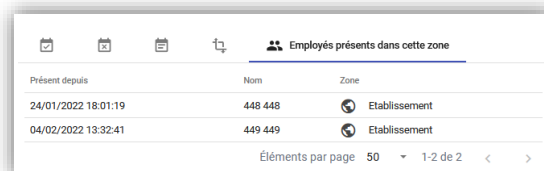


Nom	Type d'évènement	Processeur d'évènement	
PORTDUMBLEDORE Stanley (Système)	Driver Instance défaut système - SysDv	Système	☒ On
PORTDUMBLEDORE Email (Système)	Driver Instance défaut système - SysDv	Système	☒ On
PORTDUMBLEDORE NISM2 (Système)	Driver Instance défaut système - SysDv	Système	☒ On

The generators are grouped by nature (system event generators, nurse call generators, etc.) and the display shows their name as well as the type of event they generate. A switch on each line allows to activate or deactivate the corresponding generator, while a general button located at the top of the list allows to activate or deactivate all generators at once.

Displaying staff members present on zone

If the filter menu is set to the list of employees present in the zone, then the list displayed just below contains all the employees whose current presence is within the chosen zone or one of its descendants. The list is paginated and indicates when the employee was last detected in the zone.



Présent depuis	Nom	Zone
24/01/2022 18:01:19	448 448	Etablissement
04/02/2022 13:32:41	449 449	Etablissement

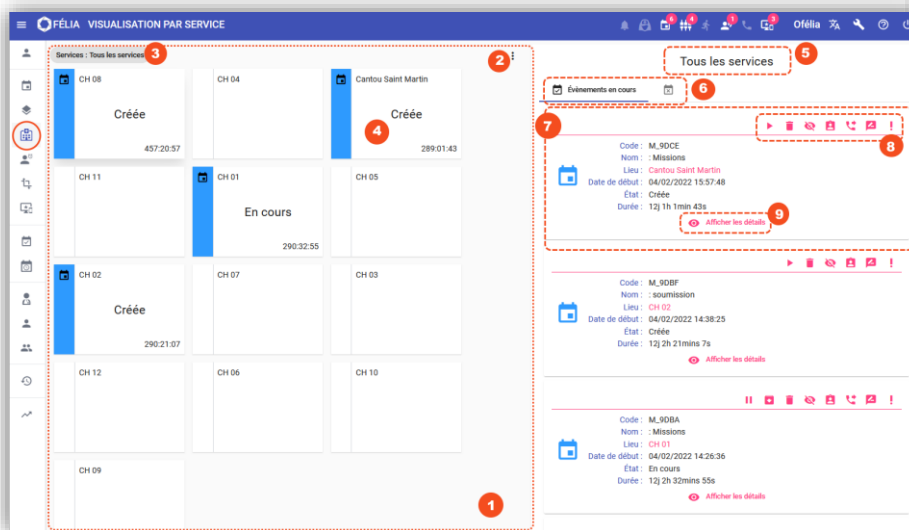
Éléments par page 50 1-2 de 2

Department / Unit View

The display of events by department, accessible directly from the side menu (), gives access to a condensed representation of the events in progress, centered on the operator's work context. The display can be reduced to only certain departments or medical units of the facility (see *Department/Unit View* on page 74), making the monitoring of a group of rooms more efficient.

The user can use filters to restrict the display to areas occupied by patients, or to those linked to a department, or to those with event generators (nurse call devices, detectors, etc.). The user can also exclude hidden events and choose to display only the zones with an event in progress or, on the contrary, have a complete view of one or more departments on which the rooms are colored when events are taking place. All the filters can be defined in the user's profile so that they are systematically applied when he/she goes to this view (see *Display Settings* on page 176).

The screen is presented in the form of a display with a grid on the left containing one box per zone (or room) and a list of current events on the right. Selecting a box representing a zone restricts the list on the right to events present in the selected zone.

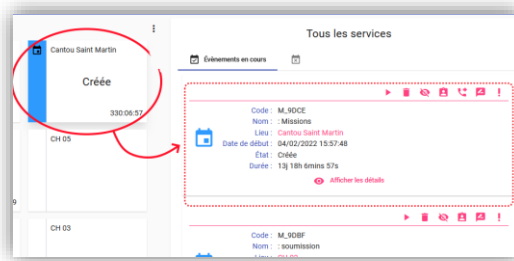


- 1 **Zone display grid** depending on the filter in place, this grid shows one card per box, each summarizing the event information for the zone (see *Zone Details* on page 190)
- 2 **Display filter**, allowing the display to be restricted by cumulative filters:
 - Only zones with event generators (nurse call devices, etc.)
 - Only zones occupied by patients (or residents, clients, etc.)
 - Only zones assigned to a department (see *New zone creation* on page 89)
 - Exclude hidden events
 - Exclude zones without events (in this case a zone appears on the screen only when an event takes place in it, and then disappears again as soon as the event is over)

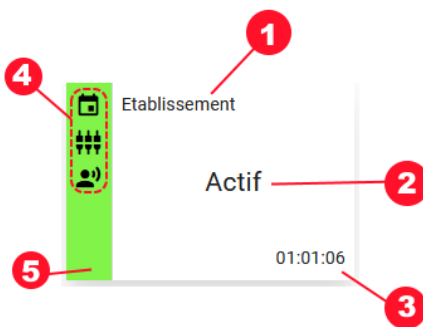
- 3 Selection (multiple) of the departments displayed, among those the user is authorized to browse (See *Roles and permissions* on page 119 and *Dynamic permissions* on page 52).
- 4 **Details Card of a zone:** See below.
- 5 **Name of the zone** currently displayed: all the events displayed in the list below take place in this zone or in one of its descendants.
- 6 **List filtering menu:** current events , hidden events.
- 7 **Event Details:** summary identical to the one in the Information area of the details window (see *General View* on page 184).
- 8 **List of Quick Actions** that can be performed on the event (see *Quick Actions* on page 183).
- 9 Link to open the **event details window** (See *Event Details* on page 184).

Zone Details

Each box in the display grid represents a summary of the events taking place in the corresponding zone, similar information is available in the list of events on the right of the screen:



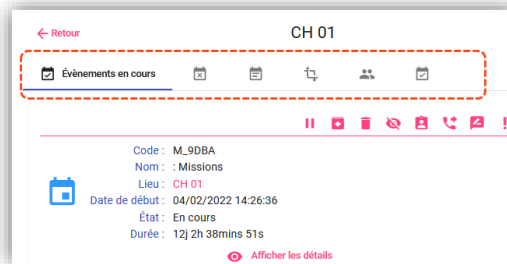
The structure of each box is as follows:



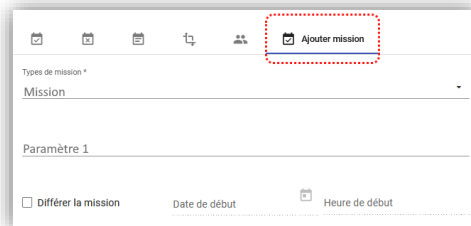
- 1 **Name** of the zone
- 2 **State** of the *main* event in the zone. The main event is defined according to the priorities set by the administrator (see *Event state definitions* on page 140)
- 3 **Time elapsed** since the creation of the main event (in hours: minutes: seconds)
- 4 **Icons** representing the natures of event currently in progress in the zone (See *Natures of events* on page 15). If there are several events of the same nature in progress, a single icon represents them.
- 5 The **color of the sidebar** corresponds to that defined by the user for the main event, and represents either its status or its type, according to the '*Applicable to map*' property of the state definition screen (See *Event state definitions* on page 140).

Selecting a zone

Whenever a zone is selected in the left-hand grid, the display focuses on the events in that zone and the top toolbar of the list changes to add additional buttons for viewing past events, event generators, as well as staff in zone. These buttons work in the same way as those in the Floor map view (see *Floor map view* on page 187).



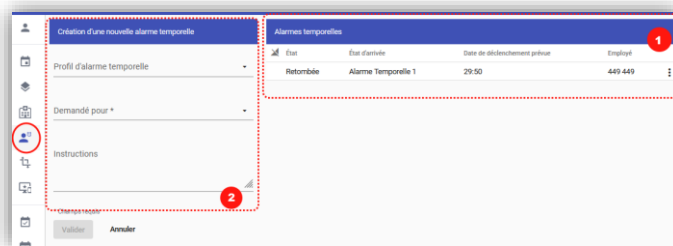
Selecting the zone also adds a new button (📅) on the toolbar, allowing the user to create a mission directly in the selected zone, by entering only its type and any associated parameters, as well as an optional delayed start date. This shortcut avoids having to go through the mission creation screen (see *Creating a mission* on page 201) if more specific information is not required:



Temporal Alarms

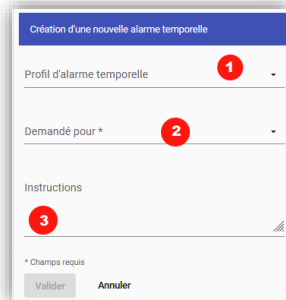
The time alarm management screen allows an authorized operator to create, deactivate, extend or cancel temporal alarms for other staff members (see *Temporal Alarms Profiles* on page 144).

The screen consists of a dashboard of the currently armed temporal alarms¹, with an area on the left² for quickly creating a new alarm. The dashboard displays for each alarm its current status, the status it will be in when triggered, the expected triggering date and the employee concerned.



For each alarm set, a contextual menu (vertical dots³) allows to cancel or extend an alarm in progress. In the latter case, the user is asked to enter how much to shift the triggering, in duration or in number of steps according to the configuration provided by the administrator (see *Temporal Alarms Profiles* on page 144).

The creation of a new temporal alarm consists of entering the following information in the quick creation zone on the left of the screen:

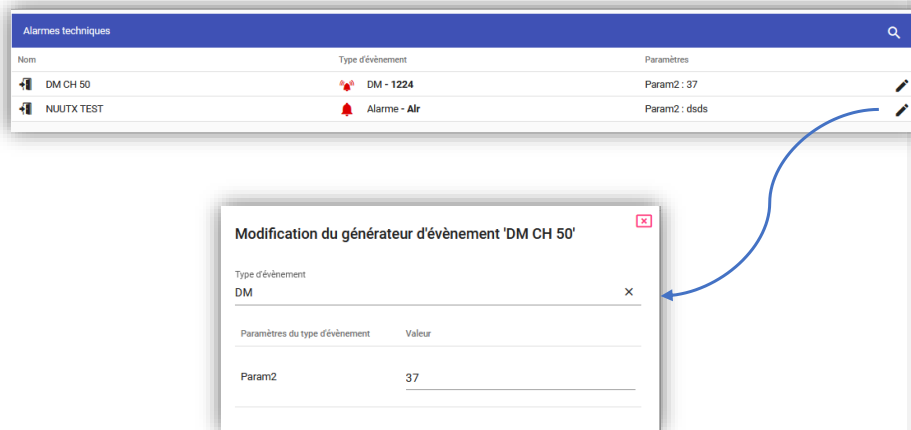


- ¹ Selection of the **temporal alarm profile** to be used, among those made available by the administrator (see *Temporal Alarms Profiles* on page 144).
- ² Selection of the **employee** the alarm will be assigned to.
- ³ **Instructions**: free text communicated to the user when the alarm is activated.

Technical Alarms

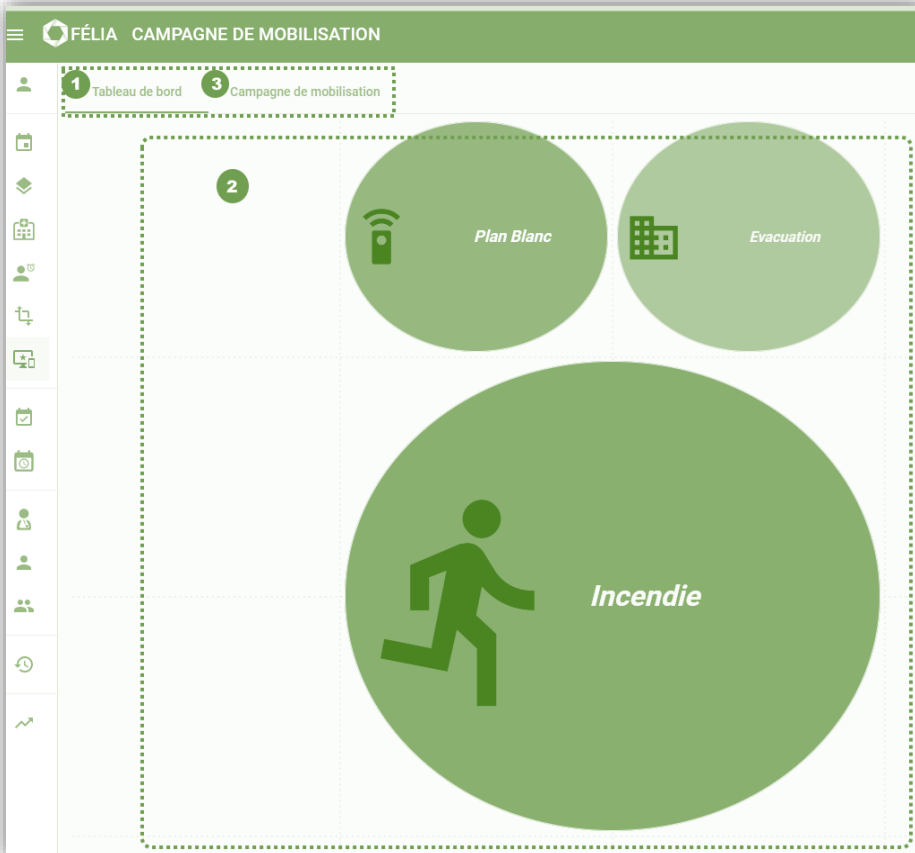
The technical alarms screen, accessible via the shortcut  in the sidebar, allows an authorized operator to edit some parameters of the technical alarm generators without having to access the administration area of the software. The configuration actions are limited to the edition of the associated alarm event type and its parameters.

The screen is setup in the form of a list of technical alarm generators, presenting for each one the type of associated alarm event as well as the parameters of this type. An icon representing a magnifying glass  allows you to search by generator name in the list, and an edit button  at the end of each line allows you to modify this information by changing the type of event or the value of its parameters:



Mobilization Campaigns

The mobilization campaigns screen, accessible via the shortcut  in the sidebar, gives an operator access to the campaign management dashboards made available by an administrator (see *Campaigns Dashboard* on page 162). The screen is composed of a main tab ¹ that contains the campaign dashboard ², and a second tab ³ that gives access to the current mobilizations.



Dashboard

From the dashboard, the operator has access to the buttons that the administrator has positioned (see *Campaign Dashboard* on page 164). A press on a button starts the corresponding mobilization campaign. Depending on the settings of the campaign, some parameters can be entered by the user: the campaign's attachment zone ¹, a comment ³, as well as the event type parameters ² (if any). A preview area at the bottom of the screen ⁴ lets the user see what the content of the different types of messages issued will be, as they are entered:

Création d'une nouvelle campagne de mobilisation

1 Campagne de mobilisation

Zone
Etablissement 1

2 Paramètres du type d'événement Valeur

Message
Merci de gagner les issues de secours au plus vite

Commentaire
3

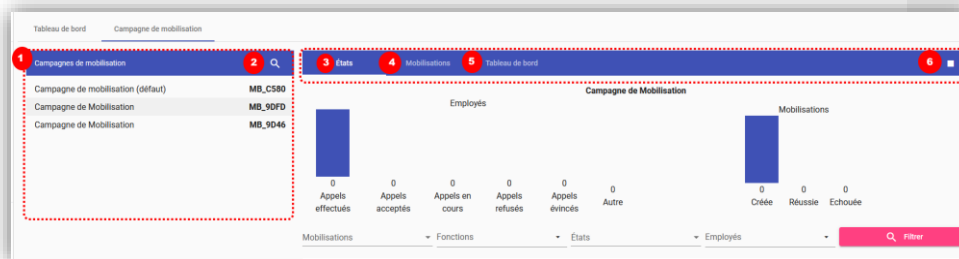
4 Aperçu

Type de message	Sujet	Message
Défaut	-	Merci de gagner les issues de secours au plus vite
email	-	Merci de gagner les issues de secours au plus vite
message court	-	Merci de gagner les issues de secours au plus vite

When the campaign information is entered and validated, the display switches to the current campaigns tab to follow the progress of the campaign that has just been initiated. If the campaign includes mobilizations configured for an automatic start, the required parameters of these mobilizations are also requested at this step.

Campaign Tracking

The campaign tracking screen presents on its left-hand side a list of current campaigns 1, in which the user can search thanks to an icon representing a magnifying glass 2, and on the right a detail area presenting the campaign selected in the list.



The details area is itself provided with 3 tabs which allow to present respectively:

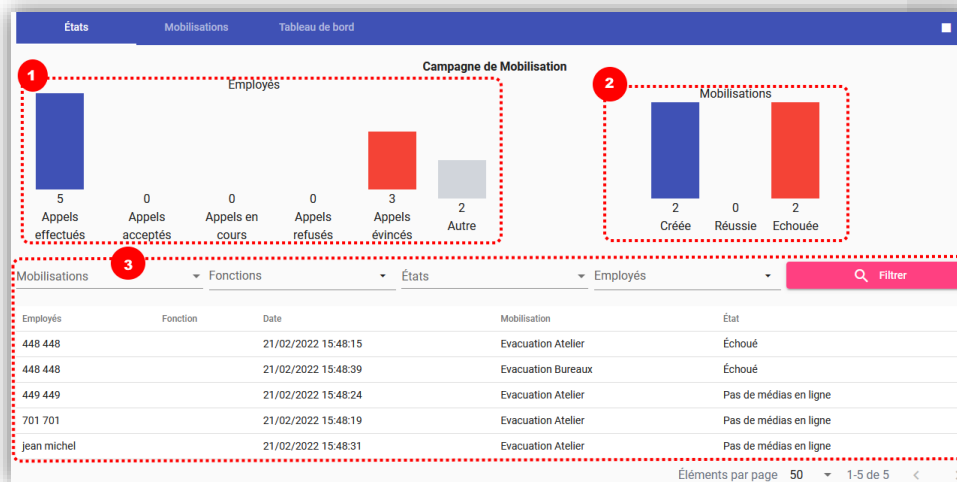
- 3 A summary of all mobilizations and a count of the number of calls already made
- 4 A follow-up of the calls for each and every mobilization
- 5 The dashboard of the mobilization campaign itself with the buttons to launch each mobilization

At the end of the tab bar is a button **6** to stop the campaign, thus stopping all mobilizations.

Campaign progress monitoring

The status tracking tab provides a summary overview of the entire mobilization campaign, call by call. All staff members called, regardless of the underlying mobilization, appear in a paginated list and several bar charts show:

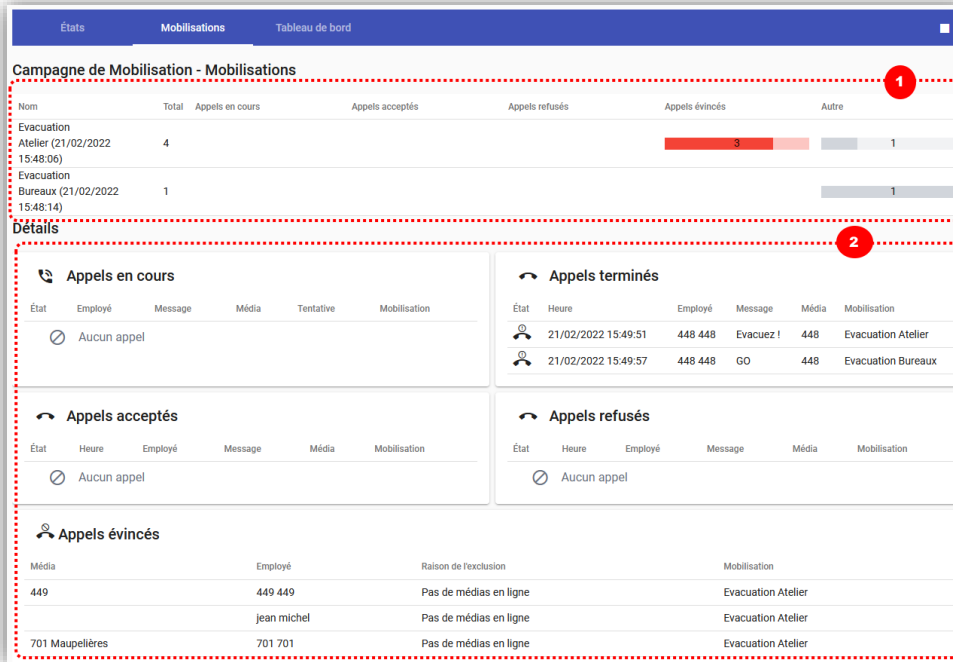
- 1 On the one hand, the overall progress of the calls: number of calls made and, among them, those accepted, in progress, refused, evicted (because no media on line or unavailable personnel for example) or other (failed calls)
- 2 On the other hand, the overall number of mobilizations created and, among them, the successful and unsuccessful ones



- 3 Finally, the bottom area of the details area lists all calls made under each engagement, along with the employee's job title, if any, the call date and the status of the call. Each column header has a drop-down list containing the unique values displayed: the user can choose one or more values from each of these lists and press the filter button to apply the chosen restriction to the whole table.

Individual mobilizations tracking

The second tab provides a more focused view of the details of calls made within each mobilization that composes the campaign.



The display is in the form of two tables placed one above the other:

- 1 A first table containing one line per mobilization and presenting for each the total number of calls made as well as the proportion of calls in progress, accepted, refused, skipped (evicted) or failed. A gauge display lets a supervisor easily measure the progress and the proportion of calls in each category.
- 1 The lower area gives the details of the history of each call in each state. This display is contextual: by default, it shows the calls of the campaign as a whole, and refocuses on the selected mobilization when one of them is clicked in the upper list.

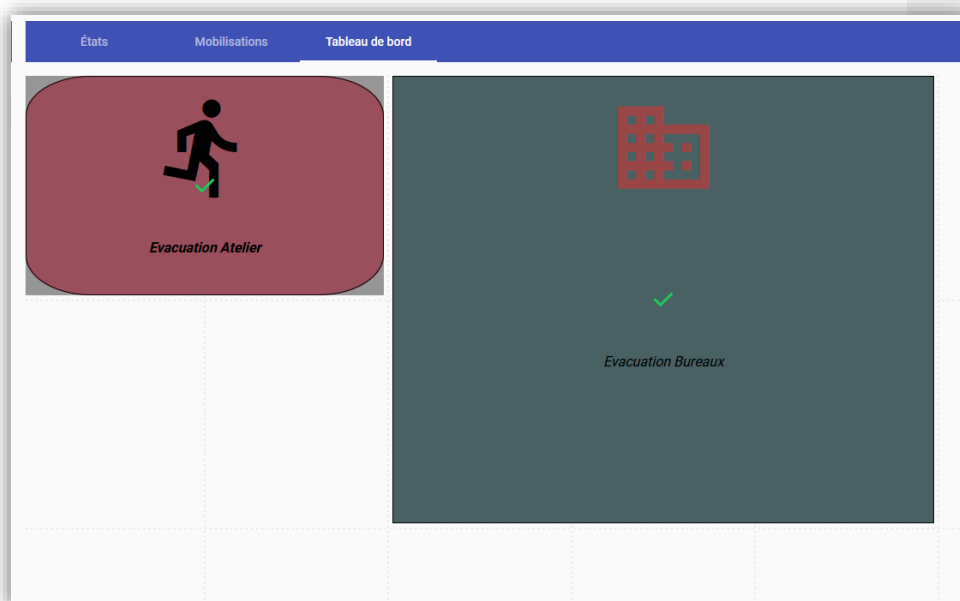
For each call, its last known state is presented. If the call is not finished, the content evolves in real time.

A simple click on a call line displays the history of all call attempts.

Appels terminés				
État	Heure	Employé	Message	Média
	21/02/2022 15:49:57	448 448	GO	448
Raison de l'appel : Sélectionné par sa fonction				
	21/02/2022 15:48:42		Appel en cours	Tentative n°1
	21/02/2022 15:49:02		Pas de réponse, attente nouvel essai - (30)	Tentative n°1
	21/02/2022 15:49:37		Appel en cours	Tentative n°2
	21/02/2022 15:49:57		Échoué	Tentative n°2

Campaign dashboard

The last tab shows the dashboard of the selected campaign, as defined by the administrator (See *Campaigns Templates* on page 60). As mobilizations are launched (automatically or by clicking on the corresponding button), their button is grayed out and a green check mark indicates that the mobilization has already been performed:



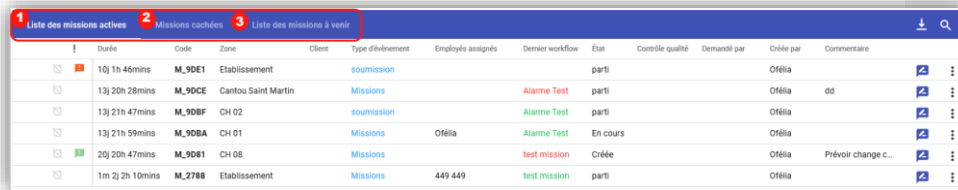
The operator can click on un-started mobilizations: they are then checked and appear in the 'Mobilisations' tab so that the user can follow their progress.

Missions

The management of missions from the operations interface meets the dual need of having a dashboard of all the missions in progress in the establishment, as well as a quick and efficient way to create new immediate or deferred missions without having to access a new window. These dedicated screens, accessible directly from the side toolbar, are particularly well suited to the logistical administration of a hotel or residential establishment, but also to the management of maintenance and service teams in any type of organization.

Missions in progress

The Current Missions screen (toolbar button ) provides access to a quick creation area on the left, as well as a Missions Dashboard in the main area. This dashboard shows the unfinished missions, grouped in three tabs:



	Durée	Code	Zone	Client	Type d'événement	Employés assignés	Dernier workflow	État	Contrôle qualité	Demandé par	Créée par	Commentaire
	1h 46mins	M_9DE1	Etablissement		soumission			parti			Ofélia	
	13j 20h 28mins	M_9DCE	Caritou Saint Martin		Missions		Alarme Test	parti			Ofélia	dd
	13j 21h 47mins	M_9DBF	CH 02		soumission		Alarme Test	parti			Ofélia	
	13j 21h 59mins	M_9DBA	CH 01		Missions	Ofélia	Alarme Test	En cours			Ofélia	
	20j 20h 47mins	M_9DB1	CH 08		Missions		test mission	Créée			Ofélia	Prévoir change c...
	1m 2j 2h 10mins	M_2788	Etablissement		Missions	449 449	test mission	parti			Ofélia	

- 1 **Active missions**, i.e., not completed, marked as visible
- 2 **Active missions** which have been **hidden** (either because of a user action or because their underlying type specifies that they should be hidden by default)
- 3 **Upcoming missions**, i.e., missions scheduled to be created at a specified time in the future.

Active Missions

The columns in the dashboard of the first two tabs described above are as follows:

- **Exceeded deadline indicator**: if a mission has exceeded the time allotted to it by an out-of-time monitor (See *Out-Of-Time* on page 138), the corresponding icon () is displayed.
- **Priority level indicator**: If the event has been assigned a priority, its icon and color are displayed (See *Event priorities* on page 140).
- **Time elapsed** since the mission was created.
- **Unique Code** generated for the mission.
- **Mission Assignment Zone**.
- **Client**: if the mission is for a customer (resp. Patient, Resident), his name is displayed.
- **Event Type** of the mission.
- **Assigned Staff**: comma-separated list of employees (first and last name) who have been requisitioned.
- **Last requisition workflow** processed for the mission.
- **Current State** of the mission (See *Missions* on page 19).

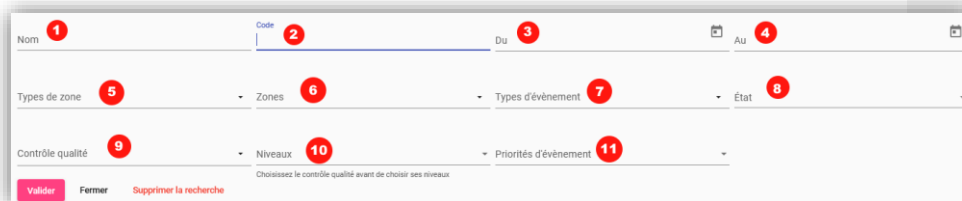
- **Quality-Of-Service:** Current level of QoS, based on the QoS monitor that has been defined as the main monitor of the missions' nature (See *Quality-of-Service Monitors* on page 135).
- **Requested by:** Name of the employee specified as the requester when creating the mission (See *Creating a mission* on page 201).
- **Created by:** name of the employee who created the mission in the software.
- **Comment:** free text entered when creating the mission.

Each line also has a button that allows you to consult the handbook of comments, or enrich it with a new time-stamped comment or an attached file. Finally, a contextual menu (vertical dots ) gives access to a list of quick actions that can be performed on the selected mission. It works in the exact same way as the *Events in Progress* dashboard (see *Quick Actions* on page 183).

Filter and export

The header of the table has on its righthand side:

- An icon allowing the contents of the table () to be exported in CSV format. The exported columns and their order correspond to the settings made in the user's profile (See *Display Settings* on page 176)
- A results filter icon () which functions in a similar way to the one in list of *Events in Progress* (see *Filters* on page 182) and whose content is shown below.



- 1 Name of the event**
- 2 Exact code** of the event to select. This option can return only one event, wildcards are not supported.
- 3 From:** start date of the event creation range.
- 4 To:** end date of the event creation range.
- 5 Zone Types** (multiple selection): For example, to see only missions taking place in rooms, corridors, etc.
- 6 Zones** (multiple selection): Limits the missions to those located in a list of zones. The zone hierarchy applies: if a zone is selected, events from that zone and all descendants of that zone will be returned.
- 7 Event Types** (multiple selection): Hierarchy applies => if an event type is selected, the events of this type and of all the descendants of this type will be returned.
- 8 State:** multi-select list of possible mission states.
- 9 Quality-Of-Service:** return only the events that have satisfied the selected QoS monitor

- ⑩ **QoS Levels** (multi-select list): If a QoS monitor is selected in ⑨, refine to return only the events that have reached one of the indicated levels.
- ⑪ **Event priorities**: multi-select list grouped by priority profile (see *Event priorities* on page 140). Returns only the events whose current priority is one of those selected.

Upcoming Missions

The upcoming missions tab shows the list of missions planned in a defined date range, according to the missions scheduled on the system. At the top is a bubble icon that contains the number of upcoming missions in the next 7 days. Just before the list of missions, there is a date range filter that serves as a filter for the list displayed:

Zone	Client	Type d'événement	Demandé par	Date de début	Type	Créée par	Commentaire
CH 02		Missions		18/02/2022 16:00:00	Différée	Ofélia	
Cantou Saint Martin		Missions		18/02/2022 18:00:00	Récurrente	Ofélia	
CH 07		soussoumission		19/02/2022 10:30:00	Différée	Ofélia	
Cantou Saint Martin		Missions		19/02/2022 18:00:00	Récurrente	Ofélia	
Cantou Saint Martin		Missions		21/02/2022 18:00:00	Récurrente	Ofélia	
Cantou Saint Martin		Missions		25/02/2022 18:00:00	Récurrente	Ofélia	

Due to the lack of progress information (duration, status, assigned employees, etc.), the columns presented in the dashboard differ slightly from those of current missions:

- **Client**: If the mission is intended for a client (resp. Patient, Resident), his name is displayed.
- **Event Type** of the mission.
- **Start date**: Scheduled start date and time for the scheduled or deferred mission.
- **Type**: **Postponed** or **Recurrent** depending on whether it is a one-off mission scheduled to start later (see *Creating a mission* below) or a mission that occurs regularly on a periodic basis (see *Scheduled Missions* on page 203).
- **Created by**: name of the employee who created the mission in the software.
- **Comment**: free text entered when creating the mission.

The rows do not give access to the buttons for manipulating the comments ledger, nor to the quick actions, but a delete icon (🗑️) allows to cancel a deferred mission or to delete a given instance of a recurring mission. In the latter case, the recurring mission itself is not modified: only the selected instance will be excluded from the series.

Creating a mission

The creation of a new mission, whether immediate or delayed, is done in the quick creation zone located on the left-hand side of the current mission screen. All the information necessary for the creation can be collected there before validation by a button located at the bottom of the panel. The required information is as follows:

Créer une mission

Zone 1

Type de mission * 2

Client 3

☐ Demande client 4

Employé 5

Priorité d'évènement 6

Commentaire 7

☐ Différer la mission 8

Date de début 9

Heure de début

* Champs requis

1 **Mission Zone.** Depending on the settings for the type of event chosen, this zone may be mandatory or optional (see *Common Properties* on page 129).

2 **Mission Type:** Type of event to create. If the type chosen has **parameters**, the list to be filled in appears here when selected.

3 **Client** (or patient, resident, etc.) for whom the mission is to be performed.

4 **Client request:** information indicating that it is the client (customer/resident/etc.) himself who has requested this mission (for example in a hotel or residential context).

5 **Employee requestor:** in the event where the mission is initiated by a specific employee.

6 **Event priority:** priority to be applied upon creation, the list is based on the priority group to which the type of event selected belongs (see *Priority groups* on page 49).

7 **Comment:** free text associated with the event, which can for example be transmitted in the message formats during requisition requests (see *Message Formats* on page 130).

8 **Postpone mission:** This checkbox allows lets the creator specify that the mission will not start immediately after validation, but rather at a specified date and time in the future.

9 **Start date and time:** in case of a deferred mission.

Scheduled Missions

The screen listing the scheduled missions (toolbar button ) opens with a list of already scheduled missions on the left (1), and a detail area to its right (2) showing the scheduling details of each selected item. The list can be filtered by name using a magnifying glass button  at the top of the list, while an add button  allows you to program a new mission. A contextual menu (vertical dots ) in the detail area allows you to modify or delete the currently selected program.



To edit a scheduled mission, first fill in the mission information in the left half of the creation window, then indicate the desired periodicity in the right half:

- 1 **Name** of the new mission.
- 2 **Mission Type**. If the selected event type has parameters, they are presented here for the user to enter.
- 3 **Mission Zone** Depending on the settings for the type of event chosen, this field may be mandatory or optional (see *Common Properties* on page 129).
- 4 **Client** (or patient, resident, etc.) for whom the mission is to be performed.
- 5 **Customer Request**: information indicating that it is the client (customer/resident/etc.) himself who has requested this mission (for example in a hotel or residential context).
- 6 **Start time**: time of creation of the mission, according to the periodicity defined in 8.
- 7 **Comment**: free text associated with the event, which can for example be transmitted in the message formats during requisition requests (see *Message Formats* on page 130).
- 8 **Periodicity** (see below).

Periodicity

Every Day

- 1 Mission creation **interval** (e.g., every 2 days, etc.).
- 2 **Start Date** of the schedule.
- 3 **Duration**: *Always* (without end date), *Until a certain date* to be specified, or for a given *number of occurrences*.

Every Week

- 1 Mission creation **interval** (e.g., every 2 weeks).
- 2 **Days of the week** when the mission is to be created (multiple selection).
- 3 **Duration**: *Always* (without end date), *Until a certain date* (to be specified), or for a given *number of occurrences*.

Every Month

- 1 Mission creation **interval** (e.g., every 6 months).
- 2 **Day of the month** by giving the number of the day or the occurrence of the day of the week (for example the Second Wednesday of the month)
- 3 **Start Date** of the schedule.
- 4 **Duration**: *Always* (without end date), *Until a certain date* (to be specified), or for a given *number of occurrences*.

Every Year

- 1 Mission creation **interval** (e.g., every 2 years).
- 2 **Day of the year** by giving the number of the day and the month or the occurrence of the day of the week and the month (for example *the Second Wednesday of January*)
- 3 **Start Date** of the schedule.
- 4 **Duration**: *Always* (without end date), *Until a certain date* (to be specified), or for a given *number of occurrences*.

Human Resources

The staff and users' management screens are a feature that sits on the borderline between administration and operations' needs. The purpose is to offer authorized users the possibility to have an insight of staff presence and to perform client and patient administration tasks without requiring full access to the software configuration area. One of the goals is to provide operational access to the list of clients and patients (depending on the license acquired and the features unlocked) allowing them to perform browsing actions, and basic editing of related information such as loyalty programs or stays. The employee screen also provides a summary of the presence and availability of the staff in the establishment.

Patients

Patient and resident management is accessible via the sidebar button  and is presented as a list of patients on the left, coupled with a detail area presenting both the characteristics of the selected patient and movement information on his or her stays in the facility. In the majority of hospital deployments, patients are synchronized through a dedicated gateway (FIHR, IHE, etc.). However, if such a gateway was not implemented (for smaller facilities, small senior residences, etc.), an add button  at the top of the list can be used to create a patient (or resident) directly in the software interface.



Creating and editing a Patient

The creation of a patient using the add button presents the same input screen as the one opened when editing the properties of a patient using the contextual menu located at the top of the detail area (vertical dots ). Since Ofelia® is not intended to be a medical record or even a care record, the required civil status information is deliberately brief. All the information that can be configured for the patient is as follows:

- 1 Patient's **Last Name**
- 2 Patient's **First Name**
- 3 **Preferred Language** (optional): in a multicultural context, provides an indication for the caregiver.
- 4 **LWP Alarm support**: Indicates whether to enable LWP alarm tracking for the patient (See note below).

- 5 **LWP Alarm Type**: Type of LWP alarm event that should be created if the patient triggers his device. If the type chosen requires parameters, these will be displayed for the user to enter.

Note: The activation of LWP alarms is only valid if the user is equipped with a compatible device, such as a medallion for example. The choice can be: *Disabled*, *Forced* (will consume a license systematically) or *Follow*

media capabilities (depending on the associated device, this will consume a license only when the device is actually active).

Stays and Movements

A patient's detail area lists the stays of the selected user. Quick action buttons allow to add or close a stay. It is not possible to have several stays open at the same time: if the patient changes department or room (or even position in a zone: bed or window, etc.) during his stay, the operator should rather create new movements to reflect the change of assignment.

Nom : Démerville Alice		Langue préférée : Fr																										
Alarme DATI : Suivre capacité du média		Type d'alarme DATI : Alarme DATI - Dati																										
Date d'entrée	Date de sortie	Zone																										
Séjours : <table border="1"> <thead> <tr> <th>Date d'arrivée *</th> <th>Heure de début *</th> <th>Zone *</th> <th>Type de position</th> <th></th> </tr> </thead> <tbody> <tr> <td>18/2/2022</td> <td>16 : 32</td> <td></td> <td></td> <td>✓ ✕</td> </tr> <tr> <td colspan="5">Sélectionnez une zone pour afficher ses types de position</td> </tr> <tr> <td>18/02/2022 16:35:00</td> <td></td> <td>Fermer</td> <td></td> <td></td> </tr> <tr> <td>28/01/2022 16:00:00</td> <td></td> <td>18/02/2022 16:32:08</td> <td></td> <td></td> </tr> </tbody> </table>				Date d'arrivée *	Heure de début *	Zone *	Type de position		18/2/2022	16 : 32			✓ ✕	Sélectionnez une zone pour afficher ses types de position					18/02/2022 16:35:00		Fermer			28/01/2022 16:00:00		18/02/2022 16:32:08		
Date d'arrivée *	Heure de début *	Zone *	Type de position																									
18/2/2022	16 : 32			✓ ✕																								
Sélectionnez une zone pour afficher ses types de position																												
18/02/2022 16:35:00		Fermer																										
28/01/2022 16:00:00		18/02/2022 16:32:08																										

Clients / Customers

The client management is accessible via the button on the side toolbar and is presented in the form of a list of clients on the left, associated with a details area presenting the information of the selected client.

Détails du client 'LeRond John'	
Nom : LeRond John	Langue préférée : en-us
Programme de fidélité : Client Gold	

The addition of a customer consists of entering his first and last names, his preferred language and his loyalty program (see *Loyalty Programs* on page 97). If this creation is automated by means of a specific driver (See *Opera Driver* on page 233), this screen allows any authorized operator to browse the list of clients present in the facility as well as the rooms they are assigned to.

Staff Dashboard

The staff dashboard screen (button on the sidebar) provides a synthetic overview of the presence and availability of the staff. The display is in the form of a grid in which each box represents an employee declared in the system. A list of all functions (i.e., jobs) is used as a filter to restrict the display of the grid to employees occupying at least one of the selected functions:

Employés			
DM		448	449
Gardiens		448	449
Soignant		448	449
	701	battling	jean
	701	joe	michel
	701 Maupellères		
	Gardiens, Soignant, DM		

- ❶ List of **functions** (jobs)
- ❷ **Staff** grid (one box per employee)

A magnifying glass icon  gives access to an additional filter on the name or the connection of the users to the graphic interface:

Employés

Nom

☐ Seulement les employés actuellement connectés

Valider Fermer Supprimer la recherche

The content of each box provides additional information on the current status of the employee:

1 Jean VALJEAN

2 448

3 Soignant, DM

4 448

5 448

6 Soignant, DM

- ❶ Employee's **First and Last Name**
- ❷ **Media Device availability** (color of the box):
■ Online, ■ Disconnected, ■ Unknown
- ❸ **Staff availability**: the icon color is the one associated with the employee's availability level (see *Availability Levels* on page 111)
- ❹ **Functions currently held**
- ❺ Employee's **Media(s) numbers**
- ❻ **Connection indicator** (to web interface)
● if currently connected

By clicking on one of the boxes, the user can send an instant message to the employee: He just has to select a media and an address in the corresponding lists as well as a call profile and the text of the message to send.

Appeler un membre du personnel

Média * 448 Adresse * 448

Message *

Ce champ est obligatoire

Profil d'appel * Default

Event history

The event history screen, accessible via the sidebar button , is similar to the *Events in Progress* dashboard, except that it only shows events that have an end date. A second tab allows you to present a specific display of nurse calls, with columns that are more relevant to this particular nature of event.

History tab

This screen displays a paginated and filterable list of completed events. This list is sorted by default to show the most recent events first, and a filter is set to display only the last 7 days. The list is of course restricted to the sole events that the current user is allowed to view.

Historique

Appels infirmière

Du : 14/02/2022

			Date de création	Date de fin	Durée	Type d'événement	État initial	Contrôle qualité	Code	Nom	Zone	Type de position	Assigné à	Demandé pour	Créé par	Commentaire
#	1		15/02/2022 10:46:54	15/02/2022 17:29:20	06:42:26	Driver Instance défaut système	Actif		S_CS3C	#PORTOUMBLEDORE Ackermann (Système) Driver Instance défaut système	Etablissement					
#			15/02/2022 10:46:46	15/02/2022 17:29:20	06:42:34	Driver Instance défaut système	Actif		S_CS3B	#PORTOUMBLEDORE Email (Système) Driver Instance défaut système	Etablissement					
#			15/02/2022 08:38:37	15/02/2022 10:46:43	02:08:06	Driver Instance défaut système	Actif		S_CS3A	#PORTOUMBLEDORE Ackermann (Système) Driver Instance défaut système	Etablissement					
#			15/02/2022 08:38:29	15/02/2022 10:46:43	02:08:14	Driver Instance défaut système	Actif		S_CS39	#PORTOUMBLEDORE Email (Système) Driver Instance défaut système	Etablissement					
			14/02/2022 10:33:45	14/02/2022 10:33:48	00:00:03	Appel infirmière	Appel		NC_CS38	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:52:26	14/02/2022 09:52:28	00:00:02	Appel infirmière	Appel		NC_CS37	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:47:29	14/02/2022 09:47:30	00:00:01	Appel infirmière	Appel		NC_CS36	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:41:29	14/02/2022 09:41:30	00:00:01	Appel infirmière	Appel		NC_CS35	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:41:15	14/02/2022 09:41:21	00:00:06	Appel infirmière	Appel		NC_CS34	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:40:35	14/02/2022 09:40:37	00:00:02	Appel infirmière	Appel		NC_CS33	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:31:44	14/02/2022 09:31:46	00:00:02	Appel infirmière	Appel		NC_CS32	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:31:22	14/02/2022 09:31:25	00:00:03	Appel infirmière	Appel		NC_CS31	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:31:11	14/02/2022 09:31:12	00:00:01	Appel infirmière	Appel		NC_CS30	# Appel infirmière	CH 44	Porte				
			14/02/2022 09:26:44	14/02/2022 09:26:46	00:00:02	Appel infirmière	Appel		NC_CS2F	# Appel infirmière	CH 44	Porte				

Éléments par page

20

61-80 de 91

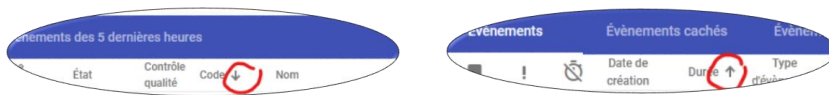
- 1 Tabs to display either the history of all events (including nurse calls) or only nurse calls.
- 2 Access to the **event handbook** to consult the time-stamped comments (read-only).
- 3 Choice of the **number of items displayed on each page** (user profile setting, see on page 176).
- 4 Access to the **next or previous page** of the list, based on the column used for sorting.

- 5 Access to the items **filter** (see *Filters* on page 210)
- 6 Export the list in **PDF format**
- 7 Export the list in **CSV format**

Columns

The rows of the history have 17 columns. Unlike the list of current events, no action is available on the events because they are all finished and it is no longer possible to act on them. A simple click on a line allows to open the details of the event (in read-only mode, without any action possible).

By clicking on the header of a column, the user can reorder the list by sorting alternatively in an ascending or descending manner by the content of this column. An icon in the table header indicates which column is used for sorting as well as the direction of the sorting.



The content of the columns in the History tab is as follows:

Column	Description
	Nature of the event represented by its icon: System Event Nurse Call Runaway LWP Alarm Alarm Mission Mobilization Campaign
	Last known priority of the event, represented by the icon and color defined by the administrator (See <i>Event Priorities</i> on page 49)
	For missions only, indicates whether an Out-of-Time monitor has been triggered (see <i>Out-Of-Time Monitors</i> on page 47) by the presence of an icon in the column
Creation date	Date and time the event was created or generated
End date	Date and time the event reached its final state
Duration	Time elapsed between the creation of the event and its end
Event Type	Name of the event type, written with the color defined for its type
Initial State	Name of the state that the event had when it was created, written with the color defined for the state (See <i>Event state definitions</i> on page 140)
Quality-Of-Service	If a default QoS of the event type applies (See <i>Quality-of-Service Monitors</i> on page 135), its level is indicated here, with its icon, and its name (in the color of the level) Trop long
Code	Unique code generated to identify the event
Name	Name generated for the event according to the formatting rule for its type (See <i>Common Properties</i> on page 129)
Zone	Last known zone of the event, i.e., its zone of attachment at creation or the last zone of presence of the fugitive in the case of a runaway
Position Type	Position within the zone
Assigned To	Requisitioned staff, as a comma-separated list of first and last names
Requested For	Requestor (patient for a nurse call, staff for an assignment, etc.)
Created By	Creator of the event when it is a manual creation
Comment	Comment entered at creation (for a mission)

Filters

By activating the magnifying glass button (🔍) at the top right of the history tab dashboard, a filter pane slides down from the top of the list and replaces the title bar. If a filter is already positioned (which it is by default on opening), the filter pane can be recalled by a down arrow button (▼). Finally, a button at the top right of this filter (✕) allows you to close it and return to the standard display. The event filtering options are as follows:

The screenshot shows a filter pane with the following elements:

- 1**: Input field for 'Code'.
- 2**: 'Du' (From) date field showing '14/2/2022'.
- 3**: 'Au' (To) date field.
- 4**: 'Natures d'évènement' (Events Natures) multiselect list.
- 5**: 'Types d'évènement' (Event types) multiselect list.
- 6**: 'État initial' (Initial States) multiselect list.
- 7**: 'Contrôle qualité' (Quality-Of-Service) multiselect list.
- 8**: 'Niveaux' (QoS Levels) multiselect list.
- 9**: 'Priorités d'évènement' (Event priorities) multiselect list.
- 10**: 'Zones' (Zones) multiselect list.
- 11**: 'Services' (Departments) multiselect list.

Below the 'Niveaux' field, there is a small text instruction: 'Choisissez le contrôle qualité avant de choisir ses niveaux'.

- 1** **Exact code** of the event to select. This option can only return one event, wildcards are not supported.
- 2** **From**: start date of the event creation range.
- 3** **To**: end date of the event creation range.
- 4** **Events Natures** (multiselect list). Only events belonging to these natures will be returned.
- 5** **Event types** (multiselect list). Hierarchy applies: if an event type is selected, the events of this type and all the descendants of this type will be returned.
- 6** **Initial States** (multiselect list), among all the states of the event natures possibly selected in **4**.
- 7** **Quality-Of-Service**: returns only those events which have satisfied the selected quality control monitor
- 8** **QoS Levels** (multiselect list). If a QoS Monitor has been selected in **7**, refine to return only events that have reached one of the selected levels.
- 9** **Event priorities**: multiselect list grouped by priority profile (see *Event priorities* on page 140). Returns only those events whose last known priority is one of those selected.
- 10** **Zones** (multiselect list): Hierarchy applies: if a zone is selected, the events of this zone and of all the descendants of this zone will be returned.
- 11** **Departments** (multiselect list): Only events that took place in a zone of these departments are returned.

Nurse Calls Tab

This tab presents a vision of the history only focused on the calls of nature 'Nurse call'. It allows you to respond to the specific need of tracking nurse calls in a hospital environment.

The operation of the screen (paginated, sortable, filterable and exportable list) is similar to that of the History tab. Only the columns displayed and the filter options differ.

Columns

The rows of the nurse call history have 10 columns, just like the history, no action is available on the events because they are all terminated and it is no longer possible to act on them. A simple click on a line gives access to the details of the event (in read-only mode, without any action possible).

The content of the columns is as follows:

Column	Description
Initial State	Name of the state the event had <u>when it was created</u> (<i>Call</i> , <i>Emergency</i> , or <i>Presence</i>), written with the color defined for the state (see <i>Event state definitions</i> on page 140)
Department	Department/Unit of the zone in which the call was triggered
Zone	Call Zone (usually a room)
Patient	Name of the patient (when known) who initiated the call
Quality-Of-Service	If a default QoS of the type of event applies (see <i>Quality-of-Service Monitors</i> on page 135), its level is indicated here, with its icon, and its name (in the color of the level)  Trop long
Start Date	Date and time the event was created or generated
Call duration	Time elapsed between the creation of the event (<i>Call</i> or <i>Emergency</i>) and a nurse or doctor presence in the room
Presence by	Staff member who used their badge to notify his presence, or type of presence associated
Presence Duration	Time elapsed between the first attendance status and the end of the last attendance
Overall Duration	Time elapsed between the creation of the event and its end

Filters

By activating the magnifying glass button () at the top right of the nursing calls tab, a filter pane slides down from the top of the list to replace the title bar. If a filter is already positioned (which is the default when opening, on the last 7 days), the filter pane can be recalled by a button representing a down arrow (). Finally, a button at the top right of this filter () allows you to close it and return to the standard display. The options for filtering nursing call events are as follows:



- 1 **From:** start date of the range of creation of the Nurse Call events.
- 2 **To:** end date of the creation range of the Nurse Call events.
- 3 **Zones** (multiselect list). Hierarchy applies: if a zone is selected, the events of this zone and of all its descendants will be returned. For example, you can enter a floor to see all the calls from the rooms on that floor.
- 4 **Departments/Units** (multiselect list). Only events that took place in a room of one of these services are returned.
- 5 **Quality-Of-Service:** only return events that have passed the selected QoS monitor.
- 6 **QoS Levels** (multiselect list): If a QoS monitor is selected in 5, refine results to return only events that have met one of the specified levels.

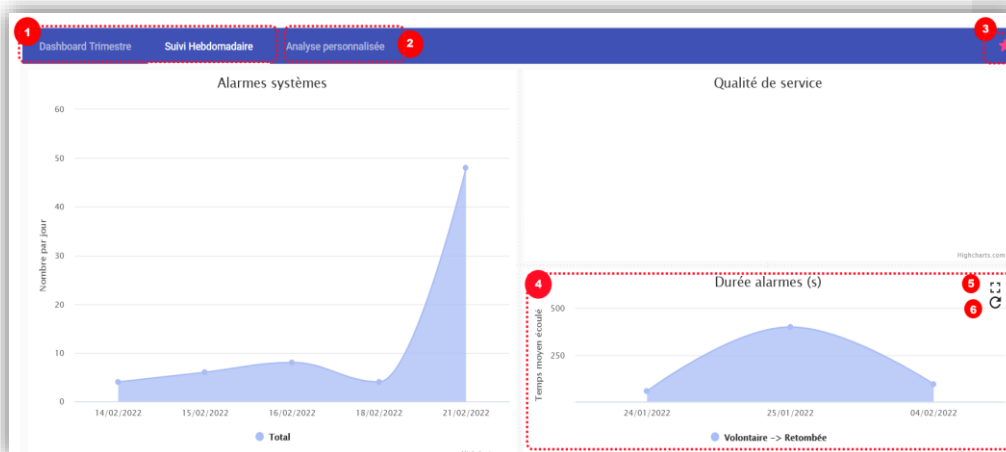
Statistics

The statistics screen, accessible via the button  on the side toolbar, provides access to the various dashboards defined by the administrator (see *Statistics Dashboards* on page 147), as well as to a graphical query tool that allows you to request a custom statistical report.

Navigation and Dashboards

The display is presented in the form of tabs with the dashboards to which the user has access (1) at the top, followed by an additional tab dedicated to personalized analyses (2), i.e., on-the-fly statistic reports not integrated into a dashboard.

When the user accesses the screen, the first dashboard configured by priority by the administrator is displayed on the screen. The user also has access to a favorite button (3) that allows him to mark a dashboard as a favorite, forcing it to be displayed on opening, regardless of the order imposed by the administrator.



Each dashboard displayed shows the various widgets (4) as defined by the dashboard designer (see *Widgets* on page 149), with the dynamic display of dates being recalculated when the screen is opened. In the upper right corner of each widget is a zoom button (5) that displays the selected widget in full screen as well as a reload button (6) that forces a recalculation of the selected widget's statistic: indeed, for performance reasons and because it is essentially a statistical tool dealing with complex aggregates of *past values*, the dashboards are **not** refreshed in real time.

It is important to note that the content of each widget rendered on the dashboards takes into account the permissions of the connected user on the underlying data: depending on their rights, two different users may therefore obtain a different graph for a same configured widget.

Custom Analysis

The last tab gives access to the wizard for calculating a custom analysis. This is to produce an on-the-fly report in a way that is very similar to the way a dashboard widget is configured, but which remains accessible to an operator who would not otherwise be authorized to edit a dashboard. The other fundamental difference is that there is no dynamic date range selection: since this is an immediate statistic that will not be replayed regularly, the dates selected are exact dates. Of course, the content of the analyzed data is always constrained to the only information that the user is allowed to browse.

Custom analyses therefore aim at executing a specific query and displaying it in the form of a graph, without this query being persisted in the system. If it is a statistic that the operator must execute regularly and that serves as a basis for his decision making, for example, it is preferable to integrate it within a dashboard widget (see *Statistics Dashboards* on page 147): it will then benefit from a pre-computation and a rendering cache and will avoid the operator to go through repetitive and tedious settings.

The parameterization is carried out by following *the three successive steps* of a wizard: *Data filtering*, selection of the *type of analysis*, and finally choice of the *data grouping and aggregation mode*. At each step a link allows you to go to the **Next** step.

Filter

The filter consists in defining the scope of the events that will be analyzed. If only the nature is selected, then all the events of this nature present in the database will be counted.

- 1 Event Nature:** The only mandatory filter. The analysis does not allow events of different nature to be mixed on the same graph.
- 2 Start Date** of the event creation range.
- 3 End Date** of the event creation range.
- 4 Event Types** (multiselect list). Hierarchical rules apply: the selection of a higher-level type involves all its descendants.
- 5 Departments/Units** (multiselect list): Limit to only those events whose main zone belongs to one of these departments.
- 6 Zone Type:** Limit to only those events whose main zone has the indicated type (e.g., Room, Hallway, etc.).
- 7 Zones** (multiselect list). Hierarchical rules apply: selecting a higher-level zone involves all its descendants (e.g., all the rooms on a floor).
- 8 Default Function/Job:** selects only those events whose type is associated with the indicated function (see *Common Properties* on page 129).

- 9 Requested by:** Selects only the events initiated by the specified employee.

Analysis Type

The choice of the type of analysis makes it possible to define what the accounting of the events will concern: it can be

- A simple **number of occurrences**,
- A **Quality-Of-Service** i.e., the number of events that have met a specific QoS or their distribution per QoS level
- A **transition duration** i.e., the time in minutes that the selected events took to transit between two specified states

The choice made in this screen determines the options that will be proposed in the next step to specify the characteristics of the chosen analysis.

2 Type d'analyse

☒ Nombre d'occurrences

☐ Contrôle qualité

☐ Durée de transition

Suivant

Grouping and aggregation

This last step allows you to choose how the data collected by the previous filters will be aggregated and grouped before being presented to the user. The information to be filled in at this stage depends on the type of analysis selected above: if, for example, the type of analysis concerns transition times, it will be necessary to specify the two states of the event between which the measurement must be made, or the QoS levels, etc.

Number of occurrences analysis

In this case, only the grouping mode and the type of aggregate are required.

The grouping options are as follows:

- **By date:** number of events created for each calendar date
- **By week:** number of events grouped by week number of creation in the year
- **By time interval:** grouping by time of creation in the day, i.e., number of events that took place between 8am and 9 am, 9am and 10 am, etc. over the period
- **By event type:** grouping of all events created for each event type, i.e., the exact type of event (even if a filter is placed on a parent event type)
- **By custom time range:** similar to the grouping by time interval, but based on the custom time ranges defined by the administrator (see *Time Ranges* on page 147): for example, morning, afternoon, etc.
- **By custom time range and date:** number of events created for each custom time range of each calendar date of the interval (for example 01/01/2022 morning, 01/01/2022 evening, 02/01/2022 morning, etc.)

3 Valider

Groupement des résultats
Par date

Type d'agrégat
Compte total

The choice of the type of aggregate available (sum, average, etc.) depends on the chosen group mode, the following table shows the different combinations:

Group Mode	Available Aggregates
By date	Total Amount
By week	Total Amount Average per hour interval Average per day
By time interval	Total Amount Average per hour interval
By event type	Total Amount Average per day Average per week Average per month
By custom time range	Total Amount Average per day
Par custom time range and date	Total Amount

Grouping of incidental event types

If the administrator has created a large hierarchy of event types, as may be the case for missions, for example, grouping them by event type can generate a wide dispersion of results that can be difficult to interpret on a graph. It is then possible to limit the display to the most frequent event types over the period (by specifying the desired number, for example "*the 5 most frequent event types*"), and by choosing whether or not to group all other occurrences in an "*Other*" category.

Quality-of-Service Analysis

The analyses related to the quality-of-service monitors (see *Quality-of-Service Monitors* on page 135) require that the monitor to be studied has been defined in advance. Because of the disparity in levels for each monitor, it is impossible to present several monitors together on the same graphical representation. Once this monitor has been selected from the list of those associated with the type of events filtered, the grouping mode and the type of aggregate for the events that have satisfied the chosen QoS monitor must be defined.

The grouping options are as follows:

- **By date:** number of events having satisfied the QoS monitor created for each calendar date
- **By week:** number of events that satisfied the QoS monitor grouped by week number of creation in the year
- **By time interval:** grouping by time of creation in the day, i.e., number of events that satisfied the QoS monitor that occurred between 8:00 and 9:00, 9:00 and 10:00, etc. over the period
- **By custom time range:** similar to the grouping by time interval, but based on the time slots defined by the administrator (See *Time Ranges* on page 147): e.g., morning, afternoon, etc.

- **By QoS level:** allows you to obtain a pie chart showing the percentage distribution between the different QoS levels achieved by the selected events.

The choice of the type of aggregate available (sum, average, etc.) depends on the chosen group mode.

The following table shows the different combinations:

Group Mode	Available Aggregates
By date	Total Amount
By week	Total Amount Average per day
By time interval	Total Amount
By custom time range	Total Amount Average per day Average per week Average per month
By custom time range and date	Total Amount

Transition duration analysis

Analysis related to the state transition durations of events requires defining which states are taken into account and the unit of measurement used (minutes, seconds, etc.). Concerning the states, only the arrival state is mandatory: if no departure state is selected, all transitions that led to the indicated state will be taken into account, regardless of their initial state. This is particularly useful for event natures with a complex state machine with several paths (see *Finite state machine* on page 18), such as Nurse Calls (see *Nurse calls* on page 19): it is therefore possible to measure the time needed to have an attendance in the room, whether the previous call was a normal call, a request for assistance or an emergency.

The grouping options are as follows:

- **By date:** average transition duration of events for each calendar date
- **By week:** average transition duration of events grouped by week number in the year
- **By time interval:** average transition duration of events by creation time in the day, i.e., number of events that took place between 8 and 9, 9 and 10, etc. over the period
- **By custom time range:** similar to the grouping by time interval, but based on the time slots defined by the administrator (see *Time Ranges* on page 147): for example, morning, afternoon, etc.
- **By custom time range and date:** average transition duration of the events for each time slot of each calendar date of the interval (e.g., 01/01/2022 morning, 01/01/2022 evening, 02/01/2022 morning, etc.).

Note: For the purposes of transition duration analysis, the aggregate type named "*Total Amount*" is assumed to be *the average duration of the transition*.

État de départ

▼

État d'arrivée *

▼

Unité de durée *

▼

Minutes

▼

Groupe des résultats

▼

Par date

▼

Type d'agrégat

▼

Compte total

▼

The choice of the type of aggregate (sum, average, etc.) available depends on the chosen group mode, the following table shows the different combinations:

Group Mode	Available Aggregates
By date	Total Amount (= average transition duration)
By week	Total Amount (= average transition duration) Average duration per day
By time interval	Total Amount (= average transition duration)
By custom time range	Total Amount (= average transition duration) Average duration per day Average duration per week Average duration per month
By QoS Level	Total Amount (= average transition duration)

APPENDIX I: Drivers' configuration

AeroScout Driver - STANLEY Healthcare®

The AeroScout Real Time Location System (RTLS ¹) is a Wi-Fi solution for monitoring the location and status of people, resources and environments.

The configuration of a driver instance requires to fill in the reserved URL and the separator used in the frames:



L'instance de driver utilisera la configuration suivante :

Url réservée *	Séparateur *
aeroscout	
Annuler	

☒ Texte
 Convertisseur de capacité ▼

The driver implements the 'Text' capability (see *Capabilities* on page 22) and can therefore be associated with a converter allowing to transform the received frames into events².

Reserved Url: This is the name of the endpoint opened on the Ofelia driver to listen to the calls. The driver will be listening on the url:

```
http(s)://OfeliaServerAddress/api/driverlisteners/ReservedUrl
```

Separator: Separation character used in text frames to split the different information (as set in AeroScout[®] system).

ASCII Driver

This driver aims to provide the ability to listen to text frames on an IP or Serial connection. As this driver has the 'Text' capability (see *Capabilities* on page 22) it can be paired with a converter in order to transform the received frames into events³. There are 3 types of connections: connection on a TCP socket, listening on a TCP socket, or connection via a Serial port. The parameters to be entered for the connection depend on its nature.

Remote Socket Connection

In this mode, the driver instance connects to a remote TCP socket whose characteristics are given in the configuration:

¹ <https://www.stanleyhealthcare.com/fr/aeroscout-rtls>

² See *Capability Converters* on page 84

³ See *Capability Converters* on page 84

- 1 **IPv4 Address** of the socket to connect to
- 2 **Port** of the connection socket
- 3 **Reconnect interval**: Time in seconds between two attempts to reconnect when the link to the socket is lost
- 4 **Heartbeat Period**: Time in seconds between 2 attempts to determine the status of the socket. If a socket is disconnected, the sign of life sent at the indicated frequency will detect this disconnection.
- 5 **Activity idle timeout**: Time in seconds for an absence of activity to be considered a disconnection.
- 6 **Connection timeout**: Time in seconds for an absence of connection to place the driver instance in failure mode.
- 7 **Start of Frame**: Character (optional) marking the beginning of the payload of a frame (by default: none, the frame is considered from its beginning)
- 8 **End of Frame**: Character marking the end of the payload of a frame (by default: new line character \r)

Local Socket Server Connection

This mode allows the driver instance to open a TCP socket listening on local addresses and to listen to frames sent by a remote application. The configuration parameters are as follows:

- ❶ **Local listening Port** for incoming connections (will be open on all available local IP addresses)
- ❷ **Error if no connection**: if this box is checked, a lack of connection should cause the driver instance to fail
- ❸ **Activity Idle timeout**: Time in seconds for an absence of activity to be considered as a disconnection.
- ❹ **Connection timeout**: Time in seconds for a lack of connection to place the driver instance in failure mode.
- ❺ **Start of Frame**: Character (optional) marking the beginning of the payload of a frame (default: none, the frame is considered from its beginning).
- ❻ **End of Frame**: Character marking the end of the payload of a frame (by default: line break character \r).

Serial Port Connection

This mode allows the driver instance to listen to frames transmitted directly on a local serial port of the *OfeliaCom*⁴ module where the driver instance is deployed. The parameters to set are those of the underlying serial protocol (EIA RS-232):

The screenshot shows a configuration window for a serial port. The 'Type' is set to 'Port série'. The fields are as follows:

- Nom * (1)
- Vitesse de transmission (baud) * (2)
- Bit de parité * (3)
- Bit de stop * (4)
- Bits de données * (5)
- ☐ DTR activé (6)
- ☐ RTS activé (7)
- Délai d'attente d'inactivité de la connexion (sec) * (8)
- Délai avant erreur (sec) * (9)
- Début de trame (10)
- Fin de trame (11)

- ❶ **Logical Name** of the serial port (for instance, COM1, COM2, etc.).
- ❷ **Transmission speed**, expressed in baud.
- ❸ **Parity bit**: Error detection calculation mode:
 - **None** (*non-parity bit frame*)
 - **Odd** (*odd parity bit*)
 - **Even** (*even parity bit*)
 - **Mark parity** (*always equal to 1*)
 - **Space parity** (*always equal to 0*)
- ❹ **Stop bit**: None, 1 bit, 2 bits, 1.5 bits
- ❺ **Number of Data bits**: depending on the number of control bits previously declared

- ❻ **DTR enabled**: indicates whether the *Data Terminal Ready* signal is used
- ❼ **RTS enabled**: indicates whether the *Data Set Ready* signal is used
- ❸ **Idle connection timeout**: time in seconds for an absence of activity to be considered a disconnection.
- ❹ **Connection lost timeout**: Time in seconds for an absence of connection to place the driver instance in failure mode.
- ❺ **Start of Frame**: Character (optional) marking the beginning of the payload of a frame (default: none, the frame is considered from its beginning).
- ❻ **End of Frame**: Character marking the end of the payload of a frame (by default: line break character \r).

⁴ See Drivers and driver instances on page 22

ASCOM IP® Driver

This driver allows to communicate with the ASCOM IP-DECT central units by means of the OAP protocol. It implements the Variable, PTI Alarm and Text capabilities (see *Capabilities* on page 22). This last capacity can be associated with a converter to convert the frames received by the OAP protocol.

- 1 **Type of connection:** OAP protocol version to use (v3 or v4)
- 2 **IPv4 Address** of the ASCOM unit
- 3 **Listening TCP Port** of the ASCOM unit
- 4 **Local listening TCP port** of the driver
- 5 Data used to reset the alarm: keyword of the remote control to send.
- 6 **System Event Type** to create in case of reset
- 7 **Localization Data Priority:** when the driver is used to transmit localization data, a drag and drop allows to order by decreasing priority the different types of localization provided.
- 8 **Driver instance Capabilities** (can be disabled individually based on actual use case)
- 9 In case of Text Capability activation, associated **capability converter** used to translate received frames.

ASCOM IP® One Way Driver

This driver is a simplified version of the ASCOM IP driver, allowing only the transmission of messages, without management of responses and acknowledgements of receipt.

L'instance de driver utilisera la configuration suivante :

Type de connexion *	Adresse *	Port *
OAP V4 1	2	1322 3

- 1** Type of connection: OAP protocol version to use (v3 or v4)
- 2** IPv4 Address of the ASCOM unit
- 3** Listening TCP Port of the ASCOM unit

DIGITAL VIRGO® Driver

This driver allows you to transmit messages in the form of SMS using the DIGITAL VIRGO®⁵ platform. It works as follows:

1. Sending SMS messages by calling a REST API on DIGITAL VIRGO®'s servers, using existing credentials
2. Receive transmission and delivery acknowledgements (ACK) in the form of an http call from the DIGITAL VIRGO® servers to a webhook listening in the driver instance
3. Receipt of response messages sent by SMS by the recipients in the form of an http call from the DIGITAL VIRGO® servers to a webhook listening in the driver instance

The configuration of the driver instance consists of setting up the outgoing connection to the DIGITAL VIRGO® API servers and the end points of the local webhook that listens to the transmission and delivery acknowledgements as well as the recipients' responses.

⁵ <https://www.digitalvirgo.com/>

Général

Adresse de l'API * **1**
<https://sms.virgopass.com/d2/multipush/>

Identifiant * **2**

Mot de passe * **3**

Accusé de réception et messages entrants

☐ Ignorer les données entrantes **4**

Adresse publique **5**
<https://serveur.ofelia.fr>

Adresse privée **6**
<http://192.168.1.70>

Temps d'attente de la confirmation (en secondes) **7**
 60

Mot clef pour l'acceptation **8**
 ACK

Mot clef pour le refus **9**
 NACK

- 1** Public Address of DIGITALVRGO® API Servers.
- 2** DIGITALVRGO® client account Id
- 3** Password for the aforementioned account
- 4** Ignore incoming data: if this box is checked, the transmission and delivery acknowledgements as well as the recipients' responses cannot be taken into account
- 5** Webhook URL to be provided to DIGITALVRGO® for transmission of acknowledgements and responses. **This must be a public address** that designates the machine hosting the driver instance (i.e., the Ofelia software itself or the OfeliaCom module)
- 6** If the OfeliaCom module is located in a DMZ, **URL of the module hosting the driver instance** as it can be resolved from Ofelia. If no DMZ or if there is a split-DNS, this URL is identical to the public address above.
- 7** Time during which the driver should **wait for a response from the user** (Accept or Reject request) before considering the call as failed
- 8** Keyword that the recipient must use in his reply SMS to indicate that he **accepts** the request
- 9** Keyword that the recipient must use in his reply

SMS to indicate that he **refuses** the request

Email Driver

This driver allows messages to be sent through an external mail server using the SMTP protocol. It authenticates itself as a user of this server and the account used is used to send messages and receive replies. It is necessary that the account used is previously configured on the SMTP server.

L'instance de driver utilisera la configuration suivante :

Adresse * **1**

Adresse SMTP * **2**

Port * **3**
 587

Identifiant **4**

Mot de passe **5**

Option de sécurité * **6**
 Automatique

- 1** Email address of the user on the server (e.g., john.doe@ofelia.fr)
- 2** SMTP address of the server: IPv4 address of the server or host name (for example mail.ofelia.fr)
- 3** TCP port of the mail server: 25, 587, etc.

- 4 **Username** of an account authorized to log in to the mail server
- 5 **Password** of the aforementioned account authorized to log in to the mail server
- 6 **Security Mode** for the connection, among:
 - Automatic (automatic negotiation)
 - None (no security)
 - SSL on connect
 - TLS
 - TLS if available

ESPA Driver

The ESPA protocol is a serial protocol commonly used in the alarm industry to interface with pagers, PBXs, etc. The transport can be established either on a TCP link (listening on a local socket or connecting to a remote socket) or on a serial link. The 'Type' parameter in the configuration screens allows to configure this transport mode.

The driver has the 'Text' capability (see *Capabilities* on page 22) and can therefore be associated with a capability converter to translate the received frames into events⁶.

TCP Connection

In case the connection is established in TCP mode (either inbound or outbound), the parameters are as follows:

L'instance de driver utilisera la configuration suivante :

Mode *	1
Maitre	
Type *	2
Connexion socket	
Adresse *	3
Port *	4 4001
Délai de transmission *	5 10
Intervalle de pooling (sec) *	6 1
☐ Activer les traces du protocole	7

- 1 **Slave Mode:** Master or Slave
- 2 **Type:** *Listen Socket* if the driver instance is waiting for a connection from an external device or *Socket Connection* if it is initiating the connection to a remote server.
- 3 **Remote IPv4 Address** if the connection type is *Socket Connection*. If the connection type is *Listening Socket*, this parameter is hidden: listening is active on all local addresses.
- 4 **TCP port number**
- 5 **Transmission timeout:** maximum time allowed in seconds to transmit a message

before considering the transmission as failed

- 6 **Polling interval:** interval in seconds between two data requests to the server (the driver instance checks periodically if new data are available)
- 7 **Enable protocol level traces:** if this box is checked the debugging information will be captured for analysis in the *Smart Inspect* tool.

⁶ See *Capability Converters* on page 84

Serial Link

In the case of a transmission mode of the ESPA protocol by a serial link, the parameters to configure for the driver instance are the following:

The screenshot shows a configuration window for a serial link. It includes fields for Mode (1), Type (2), Nom (3), Vitesse de transmission (4), Bit de parité (5), Bit de stop (6), Bits de données (7), checkboxes for DTR activé (8) and RTS activé (9), a Délai de transmission (10) field, an Intervalle de pooling (11) field, and a checkbox for Activer les traces du protocole (12).

- 1 **Slave Mode:** Master or Slave
- 2 **Type:** *Serial Port*.
- 3 **Logical Name** of the serial port (for instance, COM1, COM2, etc.)
- 4 **Transmission speed**, expressed in baud
- 5 **Parity bit:** Error detection calculation mode
 - **None** (*non-parity bit frame*)
 - **Odd** (*odd parity bit*)
 - **Even** (*even parity bit*)
 - **Mark parity** (*always equal to 1*)
 - **Space parity** (*always equal to 0*)
- 6 **Stop bit:** None, 1 bit, 2 bits, 1.5 bits
- 7 **Number of Data bits:** depending on the number of control bits previously declared
- 8 **DTR enabled:** indicates whether the *Data Terminal Ready* signal is used

- 9 **RTS enabled:** indicates whether the *Data Set Ready* signal is used
- 10 **Transmission timeout:** time in seconds for an absence of activity to be considered a disconnection
- 11 **Polling Interval:** interval in seconds between two data requests to the server (the driver instance checks periodically if new data are available).
- 12 **Enable protocol level traces:** if this box is checked the debugging information will be captured for analysis in the *Smart Inspect* tool.

HighSMS Driver

This driver allows you to transmit messages as SMS using the HighCONNECTION[®] ⁷ platform. It works as follows:

1. Sending SMS by calling a REST API on HighCONNECTION[®] servers using specified credentials
2. Receive transmission and delivery acknowledgements (ACK) as an http call from the HighCONNECTION[®] to a webhook listening in the driver instance
3. Receiving the response messages sent by SMS from the recipients as an http call from the HighCONNECTION[®] servers to a webhook listening in the driver instance

⁷ <https://www.highconnexion.com/>

The configuration of the driver instance consists of setting up the outgoing connection to the HighCONEXION® API servers, and the endpoints of the local webhook that listens for transmission and delivery acknowledgements, and for the recipients' responses.

The image shows a configuration form for Ofelia. It has several input fields and a checkbox, each with a red circle containing a number from 1 to 8. The fields are: 'Url du Serveur *' with value 'https://highpush-v50.hcnx.eu' (1); 'Identifiant *' (2) and 'Mot de passe' (3); 'Url publique du serveur local *' with value 'ofelia.cloud-ascom.eu' (4); 'Mot clef pour l'acceptation *' with value 'ACK' (5) and 'Mot clef pour le refus *' with value 'NAK' (6); 'Expéditeur *' with value 'Ofelia' (7); and a checkbox 'Traces Http' (8).

- ❶ **Server URL:** URL to reach the HighCONEXION® server.
- ❷ **Username** provided by HighCONEXION® on their platform.
- ❸ **Password** of the aforementioned HighCONEXION® account.
- ❹ **Local server Public URL:** host name (FQDN) of the server hosting the driver (Ofelia server in single-server mode) preceded or not by the protocol indicator http:// or https://. If only the hostname is specified, the driver will opt for an unsecured http connection. This name must be publicly resolvable from the HighCONEXION® servers.
- ❺ **Keyword** that the recipient must use in his reply SMS to indicate that he **accepts** the request.
- ❻ **Keyword** that the recipient must use in his reply SMS to indicate that he **refuses** the request.
- ❼ **Sender:** Name sent as the sender of outbound SMS messages.
- ❽ **HTTP Traces:** If this box is checked additional diagnostic information about HTTP exchanges is sent to the *Smart Inspect* tool.

HTTP Driver

This driver implements a mini local web server able to receive text frames in the form of GET requests. The driver has the 'Text' capability (see *Capabilities* on page 22) and can therefore be associated with a capability converter to translate the received frames into events ⁸.

Url réservée *	Séparateur *
basic 1	2

MiniFinder® Driver

This driver is used to track MiniFinder® ⁹GPS beacons. The configuration of the driver instance consists only in setting the TCP port on which the connection should be established.

L'instance de driver utilisera la configuration suivante :

Port *

35062

MODBUS® Driver

The Modbus® protocol is an open communication protocol designed to make industrial equipment communicate, based on a master/slave architecture. The protocol is primarily intended to enable simple, reliable and fast communication between automation and field devices. A single Modbus® client device initiates information requests by sending requests to one or more Modbus® server devices on the same network.

The connection can be made over a TCP socket (listening on a local socket or connecting to a remote socket) or over a serial link. The 'Type' parameter in the configuration screens allows to configure this transport mode.

⁸ See *Capability Converters* on page 84

⁹ <https://minifinder.com/>

TCP Connection

In case the connection is established in TCP mode (either inbound or outbound), the parameters are as follows:

1 **Type:** *Listening Socket* if the driver instance is waiting for a connection from an external device or *Remote Socket Connection* if it is initiating the connection to a remote server.

2 **Remote IPv4 Address** if the connection type is *Socket Connection*. If the connection

type is *Listening Socket*, this parameter is hidden: listening is active on all local addresses.

3 **TCP port number**

4 **Transmission timeout:** Maximum time allowed in seconds to transmit a message before considering the transmission as failed

5 **Parameters:** See below.

Serial Link

In case the MODBUS transmission is achieved through a serial link, the parameters required to configure the driver instance are the following:

1 **Type:** *Serial Port*

2 **Logical Name** of local serial port (COM1, COM2, etc.)

3 **Transmission speed**, expressed in baud

4 **Parity bit:** Error detection calculation mode

- **None** (*non-parity bit frame*)
- **Odd** (*odd parity bit*)
- **Even** (*even parity bit*)
- **Mark parity** (*always equal to 1*)
- **Space parity** (*always equal to 0*)

5 **Stop bit:** None, 1 bit, 2 bits, 1.5 bits

6 **Number of Data bits:** depending on the number of control bits previously declared

7 **DTR enabled:** indicates whether the *Data Terminal Ready* signal is used

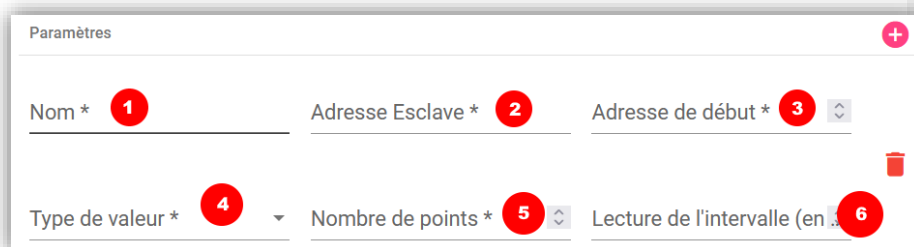
8 **RTS enabled:** indicates whether the *Data Set Ready* signal is used

9 **Transmission timeout:** time in seconds for an absence of activity to be considered a disconnection

10 **Parameters:** See below

Parameters

Whatever the selected transmission mode, the driver configuration allows to define an unlimited number of address parameters indicating the address offsets where the driver can read values. The addition of a parameter is done by the add button  and requires the following information:



- ❶ **Parameter Name**
- ❷ **Address** of the slave PLC
- ❸ **Start Address** of the reading range
- ❹ **Value Type:**
 - Output Register (16bits)
 - Input Register (16bits)
 - Discrete value (output)
 - Discrete value (input)
- ❺ **Number of points** to read from the beginning of the range
- ❻ **Poll interval:** time in seconds between two data acquisitions (polling)

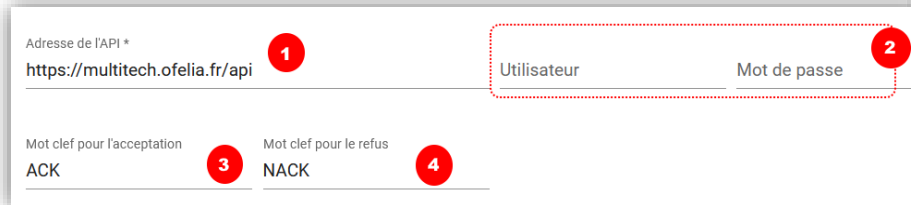
MultiTech® Driver

This driver allows the use of a GSM modem of the *MultiConnect® Cell 100*¹⁰ line to send SMS messages. Sending is performed through an HTTP API integrated in the modem. The configuration of the connection consists in indicating the address of the API as well as the authentication information.



The activation of the API and the creation of the user account are a prerequisite to be done directly in the administration interface of the MultiTech® modem. The transmission and delivery information, as well as the recipients' responses, are routed through the MultiTech® API and transmitted to Ofelia® for requisition tracking. The configuration of the driver instance allows you to choose the keywords that will be used by the recipients to accept or refuse a requisition. The configuration is as follows:

¹⁰ <https://www.multitech.com/brands/multiconnect-cell-100-series>



Adresse de l'API *

https://multitech.ofelia.fr/api

Utilisateur

Mot de passe

Mot clef pour l'acceptation

ACK

Mot clef pour le refus

NACK

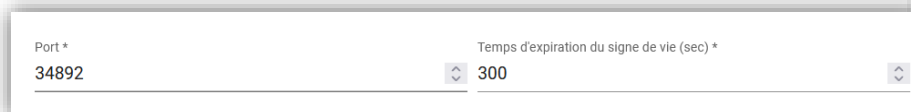
- 1 **API Address**, reachable from the driver instance location
- 2 **User credentials**. The account must have been provisioned and authorized in the interface provided by MultiTech®
- 3 **Keyword for acceptance**: If this keyword is sent by a user in response to a requisition message, the requisition will be considered as **accepted** by the user.
- 4 **Keyword for refusal**: If this keyword is sent by a user in response to a requisition message, the requisition will be considered as **rejected** by the user.

Virtual Nurse Call Driver

This driver represents a virtual nurse call box, used internally for control. Once created on an *OfeliaCom* module, it does not require any additional configuration.

Ofelia Messenger Driver

This driver allows to establish a communication channel with the *Ofelia Messenger* notification applications running on *Microsoft Windows*® workstations on the local network. Its configuration is necessary in order to communicate with these applications through a TCP socket connection. Setting it up simply consists in establishing on which port the application must connect as well as the heartbeat frequency:



Port *

34892

Temps d'expiration du signe de vie (sec) *

300

The heartbeat is a signal sent at regular intervals (defined above) to detect if the application is still connected to the server.

OfeliaSmart Driver

This driver allows to establish a communication channel with Ofelia Smart mobile applications running on Android[®] smartphones. Its addition is necessary to communicate with these applications through a TCP socket connection. Its configuration consists in establishing on which port the device should connect and the frequency of sending a sign of life used to detect disconnections of mobile devices.

Port *	Temps d'expiration du signe de vie (sec) *
38240	300

The application has the ability to geolocate the device in different ways (by scanning an NFC tag, a barcode, Bluetooth[®] beacons, GPS, Wi-Fi access points, etc.). Through a drag-and-drop operation, the user can configure the priority of the locations by ranking the different means used in decreasing order of reliability (the most reliable at the top of the list, to end with the most approximate).

Priorités des localisations	
Nfc	
Barcode	
Ble	
Gps	
Wifi	

OPC UA Driver

OPC Unified Architecture, aka *OPC UA*, is a cross-platform, open source IEC62541 standard for data exchange between sensors and cloud applications developed by the OPC Foundation. The protocol is based on a client-server architecture, and the elements to configure on an instance of this driver are:

Adresse *	☐ Utiliser la sécurité
Période de reconnexion (en seconde) *	☐ Désactiver la vérification du domaine
10	

- ❶ **Address:** OPC Server Address
- ❷ **Reconnection period:** Delay before trying a new connection attempt in case of link loss
- ❸ **Use secure mode:** If this box is checked, TLS/SSL protocol will be used
- ❹ **Disable Domain verification:** By default, the protocol checks the FQDN validity on the certificate, if this box is checked this verification is omitted.

Opera Driver

OPERA® software from Oracle Hospitality® (formerly Micros®) is the largest PMS¹¹ player in the hotel industry. The driver integrated in Ofelia® allows to set up a one-way gateway for synchronizing guests and stays. The objective is to provide Ofelia® software with movements and room assignments to customers.

The connection to the Opera® server is done on a TCP socket and the configuration of an instance of this driver is as follows:

- 1 **IPv4 Address** of Opera® server
- 2 **Remote TCP Port** of the Opera® server
- 3 **Heartbeat**: Data retrieval interval (in seconds)
- 4 **Encoding** of messages sent by Opera®: due to the different possible remote configurations, a set of ISO encodings is proposed for selection. It is advisable to test the correct integration of messages with different extended characters (accents and other diacritical signs, etc.) when setting up the system.
- 5 The software allows you to assign **a title and a language** to each customer in the form of a free text. As this information is critical in customer relations (especially in the luxury hotel industry), it is possible to add **+** as many translations as desired by adding a language code issued by Opera® and a title for a man, a woman or a couple to the list. For example:

Language	Male	Female	Couple
FR	M.	Mme	M. et Mme
EN	Mr.	Mrs.	Mr. & Mrs.

¹¹ *Property Management System*

Ping Driver

This driver allows to check the connectivity of a network event by sending an ICMP request at regular intervals.

Intervalle de temps en cas de réussite (sec) *	Intervalle de temps en cas d'échec (sec) *
60	10

- ❶ **Time Interval on Success:** time in seconds between two PING attempts if the remote host has responded
- ❷ **Time Interval on Failure:** time in seconds before the next PING attempt if the host did not respond

Pop-up Driver

This driver allows message transmission to Windows® clients compatible with the Mercury® notification application. It is integrated into Ofelia® for compatibility with the existing installed base, and is intended to be replaced by the Ofelia Messenger notification application. Its implementation only requires configuring the TCP connection port used by the application on the client workstations.

Port de connexion *
38420

SIA® Driver

The SIA protocol is an exchange protocol implemented by the *Security Industry Association* (SIA¹²) that aims to standardize communications and make security equipment from different vendors compatible and interchangeable. It is therefore mainly used in the world of industrial alarms. The connection is a TCP or UDP datagram and supports 3 different message formats:

- **SIA-DCS:** *SIA DC-04* format
- **ADM-CID:** *Contact ID DC-05* format
- **CUSTOM ADM-CID:** *Contact ID DC-05* format with a custom message header

¹² <https://www.securityindustry.org/>

Whatever the message format chosen, the configuration relative to the SIA driver instances requires setting up the connection to the equipment as well as general elements of the communication mode (encryption, supervision, number of attempts, etc.).

Main Settings

The main parameters of the driver instance concern the way of establishing the connection with the system: it is necessary to fill in the account identification number as well as its prefix and the receiver number. It is also necessary to indicate whether the connection to the alarm station is made in connected mode (TCP protocol) or datagram mode (UDP protocol), as well as the IPv4 address of the station and its connection port.

The screenshot shows a configuration window titled 'Paramètres généraux'. It has two sections: 'Station' and 'Port *'. Under 'Station', there are fields for 'Numéro de compte *' (containing '12345A'), 'Préfixe de compte', and 'Numéro de récepteur'. Under 'Port *', there is a dropdown menu for 'Mode de connexion *' (set to 'UDP') and a text field for 'Adresse *'. The 'Port *' field contains '9401'.

Account Number: (#acct) This account number consists of the "#" character, followed by a hexadecimal representation of the account number in the form of 3 to 16 ASCII characters.

Account prefix: (LprRef) This prefix consists of the "L" character, followed by a hexadecimal representation of the account number in the form of 1 to 6 ASCII characters.

Receiver number: (Rrcvr) This prefix consists of the character "R", followed by a hexadecimal representation of the receiver number in the form of 1 to 6 ASCII characters.

Communication

The details of the communication mode with the alarm control panel are set by means of the following properties:

The screenshot shows a configuration window titled 'Communication'. It has several settings: 'Délai de réponse (sec)' (20), 'Nombre de tentatives' (3), 'Activer la supervision' (checked), 'Intervalle de supervision (sec)' (3), 'Activer le cryptage' (checked), 'Taille de la clé' (32), 'Clé' (4), and 'Inclure l'horodatage' (checked). Red circles with numbers 1 through 5 are placed next to the following settings: 1. Délai de réponse (sec), 2. Nombre de tentatives, 3. Activer la supervision, 4. Clé, 5. Inclure l'horodatage.

1 Delay in seconds before retrying if no response

2 Number of attempts to be made before considering a transmission as definitely failed

3 Enable supervision: if this box is checked, a sign of life will be sent at the specified interval (in seconds) to check that the control panel is still reachable

4 Enable Encryption: if this box is checked, the exchanges will be encrypted in AES128, AES192 or AES256 depending on the selected key size. The encryption key is required, it must be entered in hexadecimal format and have a length according to the chosen algorithm (32, 48 or 64 hexadecimal characters respectively). If encryption is requested, the timestamp is automatically checked and mandatory.

- 5 **Include Timestamp**: Forces the addition of a timestamp (GMT) to the message. This parameter is mandatory if encryption is enabled, optional otherwise.

Messages Content

Depending on the message format chosen, the content of the messages must be configured so that each possible value of the source protocol's event codes corresponds to a change in event status in Ofelia®.

SIA-DCS Format

The SIA-DCS format is the richest: it allows to manage both the beginning and the end of an alarm (*Restore*). The configuration consists in associating to each SIA code (on 2 letters) one of the corresponding status changes in Ofelia. The event beginnings and ends that can be configured in Ofelia are the following: Technical alarm creation or fallback, LWP alarm creation or fallback, or nurse call creation or end. The default setting is as follows:

The screenshot shows a configuration window titled "Paramètres des messages". It contains several fields for configuring SIA-DCS messages:

- Format ***: A dropdown menu set to "SIA-DCS".
- Code d'événement d'alarme technique ***: A text field containing "HA - Holdup Alarm".
- Code d'événement de disparition d'alarme technique ***: A text field containing "UH - Untyped Alarm Restore".
- Code d'événement d'alarme dati ***: A text field containing "PA - Panic Alarm".
- Code d'événement de disparition d'alarme dati ***: A text field containing "PH - Panic Alarm Restore".
- Code d'événement d'appel infirmière ***: A text field containing "MA - Medical Alarm".
- Code d'événement de disparition d'appel infirmière ***: A text field containing "MH - Medical Alarm Restore".

ADM-CID (Contact Id) Format

The Contact-ID format only allows the creation of events, with no control over their termination. The configuration consists in associating to each Contact ID code (3 digits) one of the corresponding status changes in Ofelia: Technical alarm creation, PTI alarm creation, or nurse call creation. The configuration also allows to assign a group number to the message, the default values are as follows:

ADM-CID Custom Format

The Custom ADM-CID mode is identical in all respects to the previous one (Contact ID) but it also allows the entry of a personalized message header to identify the frames concerned.

SIP Driver

SIP (*Session Initiation Protocol*) is an open standard communication protocol for session management often used in multimedia telecommunications (sound, image, etc.). The driver integrated into Ofelia allows voice messages to be sent using VoIP (Voice over IP). To use the VoIP features of this driver, the Ofelia text-to-speech service must be installed and correctly started on the server.

The configuration of the driver instance requires the following settings:

- ① **Proxy Address:** Address of the IPBX unit

- 2 **Port:** Local listening port on the driver
- 3 **Name:** Name of the SIP subscriber Ofelia impersonates
- 4 **Identifier:** SIP subscriber Id. If no identifier is provided, the name will be used.
- 5 **Password:** Password for the SIP subscriber (see above).
- 6 **Domain:** Subscriber's domain name (i.e., the part after the "@" character)
- 7 **Use registration:** If this box is checked, the *SIP registrar* is used
- 8 **Display name:** Name presented in outbound calls
- 9 **Linked address:** local IPv4 address used for the outbound connection to the IPBX unit
- 10 **Public address:** local IPv4 address used to receive answers from the IPBX unit
- 11 **Use IP address in FROM field:** if this box is checked the FROM field of the SIP protocol will be filled with the public address
- 12 **NAT rewrite for SDP:** Alcatel OmniPCX Office (OXO) specific setting.
- 13 **Text-to-speech engine settings:** Utterance speed and volume.

SMS Factor® Driver

This driver allows you to transmit messages in the form of SMS using the **SMSFactor®**¹³ platform. It works as follows:

1. Sending SMS messages by calling a REST API on **SMSFactor®**'s servers, providing login information in the form of an API token
2. Receive transmission and delivery acknowledgements as an http call from the **SMSFactor®** servers to a webhook listening in the driver instance
3. Receipt of the response messages sent by SMS by the recipients in the form of an http call from the **SMSFactor®** servers to a webhook listening in the driver instance

The configuration of the driver instance consists in setting up the outgoing connection to the **SMSFactor®** API servers on the one hand, and on the other hand the endpoints of the local webhook that listens to the transmission and delivery acknowledgements and the recipients' replies.

- 1 **SMSFactor®** API's **public address**
- 2 **Token:** authentication token for an **SMSFactor®** client account
- 3 **Ignore incoming data:** if this box is checked, the transmission and delivery acknowledgements as well as the recipients' replies cannot be taken into account
- 4 **Webhook URL** to be supplied to **SMSFactor®** so that it can transmit acknowledgements and replies. **It must be a public address** that designates the machine holding the driver instance (i.e., the Ofelia software itself or the *OfeliaCom* module hosting this driver instance)
- 5 If the *OfeliaCom* module is located in a DMZ, this is the **URL of the module hosting the driver instance** as it can be resolved from Ofelia. If no DMZ or if there is a split-DNS, this URL is identical to the public address above.

¹³ <https://www.smsfactor.com/>

- ⑥ Time during which the driver must **wait for a response from the user** (*Acceptance* or *Refusal* of requisition) before considering the call as failed.
- ⑦ **Keyword** that the recipient must use in his answer SMS to indicate that he **accepts** the request
- ⑧ **Keyword** that the recipient must use in his answer SMS to indicate that he **refuses** the request

SMTP Server Driver

This driver behaves like a local SMTP server and allows to receive text frames in the form of electronic messages. As it has the 'Text' capability (see *Capabilities* on page 22) it can therefore be associated with a capability converter allowing to generate events from emails sent by a user or an automaton. The configuration of an instance of this driver requires the following parameters:

The screenshot shows a configuration form for the SMTP Server Driver. It includes fields for 'Nom *' (1), 'Port *' (2) with a dropdown arrow, a checkbox 'Utiliser SSL' (3), a checkbox 'Utiliser l'authentification' (4), 'Identifiant' (5), and 'Mot de passe' (6). A red error message 'Ce champ est obligatoire' is visible below the 'Nom *' field.

- ① **Local host name**
- ② Local **connection Port** to SMTP service (25, 587, 465, etc.)
- ③ **Use SSL**: If this box is checked, the connection will require the use of SMTPS (SMTP over SSL)
- ④ **Use authentication**: if this box is checked, the connection to the SMTP server will require authentication (*Basic Auth*) with the provided **username** (⑤) and **password** (⑥).

Stanley® Driver

This driver can be interfaced with the LWP alarm devices provided by STANLEY Security®. These devices have 2 alarm levels (code Red and code Blue) and a presence/attendance state (equivalent to a reset alarm in Ofelia's LWP alarm concepts).

The configuration of an instance of this driver consists in indicating the name of the route (reserved URL) and in matching each of these 3 states with their equivalent in Ofelia® LWP events:

The screenshot shows a configuration form for the Stanley Driver. It has two rows of dropdown menus. The first row maps 'Url réservée *' (stanley) to 'État RedAlarm *' (Volontaire). The second row maps 'État BlueAlarm *' (Acquittée) to 'État Present *' (Retombée).

Reserved URL: This is the name of the endpoint opened on the Ofelia driver listening to the calls. The driver will be listening on the URL:

`http(s)://OfeliaServerAddress/api/driverlisteners/ReservedURL`

teleCARE® IP Driver

This driver is specific to the proprietary system for managing sick calls and runaways provided by ASCOM®. The driver is designed to connect in IP to an ASCOM® central unit in an authenticated manner and to listen to messages sent in teleCARE® IP format from nurse call devices or medallions.

The driver has the 'NurseCall' capability (see *Capabilities* on page 22) and can therefore be associated with a capacity converter in order to translate nurse calls into other events¹⁴. The information to enter to set up an instance of this driver is as follows:

Adresse * 1	Port * 2 2222
Identifiant 3	Mot de passe 4
Mot clé de fugue * 5 FUGUE	Mot clé de batterie basse * 6 Battery alarm
☒ Ignorer les fin de fugues 7	

- 1** IPv4 Address of ASCOM® unit
- 2** Connection Port on the unit
- 3** Identifier of a user declared in ASCOM® unit's control panel
- 4** Password of the aforementioned user
- 5** Keyword set up in ASCOM® unit to notify a runaway
- 6** Keyword set up in ASCOM® unit to notify a low battery alert

¹⁴ See *Capability Converters* on page 84

teleCARE® M Driver

The teleCARE® system is a networked hospital communication system for hospitals and other healthcare facilities. It is modular, scalable and designed as a *Local Operated Network* (LON) consisting of an unshielded twisted pair of wires. The interface to the LON bus is an open-architecture transceiver (*FTT*), which guarantees galvanic isolation and provides a flexible, reliable and easy-to-install LON bus. The teleCARE® system allows integration with the ASCOM® communication platform for message exchange and with the ASCOM® 9d DECT system for mobile communication. The driver implements a '*NurseCall*' capability (see *Capabilities* on page 22) and can therefore be combined with a capability converter to translate nurse calls into other events¹⁵.

The transport of information by the protocol can be done either by means of UDP datagrams (sent by the ASCOM® central unit on a socket of the driver), or by a serial link. The chosen transport mode influences the connection and transport information to be set.

UDP Sockets

When using datagrams for the communication with the central unit, the information to be provided concerns the local port to which the central unit sends the data frames. The driver listens on this port on all free addresses. A second parameter allows to set a maximum delay (in seconds) to detect an inactivity when no data is received.



The screenshot shows a configuration window for the teleCARE M Driver. It has three fields with labels and asterisks indicating they are required:

- Type ***: Set to "Socket UDP".
- Port ***: Set to "15001".
- Délai d'attente d'inactivité de la connexion (sec) ***: Set to "0".

Each field has a small dropdown arrow on the right side.

Serial Link

In case the connection between the driver and the ASCOM central unit takes place through a serial link, the information is more numerous and aims at defining precisely the binary exchange protocol used:

¹⁵ See *Capability Converters* on page 84

The screenshot shows a configuration window for a serial port. It includes fields for 'Type' (set to 'Port série'), 'Nom', 'Vitesse de transmission (baud)', 'Bit de parité', 'Bit de stop', 'Bits de données', checkboxes for 'DTR activé' and 'RTS activé', and a 'Délai d'attente d'inactivité de la connexion (sec)' field set to 30. Red circles with numbers 1 through 9 are placed over specific elements: 1 on 'Type', 2 on 'Nom', 3 on 'Vitesse de transmission', 4 on 'Bit de parité', 5 on 'Bit de stop', 6 on 'Bits de données', 7 on 'DTR activé', 8 on 'RTS activé', and 9 on the timeout value '30'.

- 1 **Type:** *Serial Port*
- 2 **Logical Name** of local serial port (COM1, COM2, etc.)
- 3 **Transmission speed**, expressed in baud
- 4 **Parity bit:** Error detection calculation mode
 - **None** (*non-parity bit frame*)
 - **Odd** (*odd parity bit*)
 - **Even** (*even parity bit*)
 - **Mark parity** (*always equal to 1*)
 - **Space parity** (*always equal to 0*)
- 5 **Stop bit:** None, 1 bit, 2 bits, 1.5 bits
- 6 **Number of Data bits:** depending on the

number of control bits previously declared

- 7 **DTR enabled:** indicates whether the *Data Terminal Ready* signal is used
- 8 **RTS enabled:** indicates whether the *Data Set Ready* signal is used
- 9 **Transmission timeout:** time in seconds for an absence of activity to be considered a disconnection

Common Settings

Whatever the type of network transport chosen, it is necessary to configure for each driver instance the correspondence between the keywords received in the teleCARE[®] frames and the event states associated with them in Ofelia[®]. The administrator can configure which characters will be used to identify the beginning and the end of the frame and associate each protocol keyword with its equivalent in the events:

Début de trame	Fin de trame
	\n
Mot clé	
Appel	Assistance
app	ass
Urgence	Présence
urg	pres
Toilette	Fin d'appel
toilet	off
Fugue	Mot clé de batterie basse
fugue	pile basse
☒ Ignorer les fin de fugues	

A checkbox at the end of the configuration also allows you to never process runaway ends.

APPENDIX II – External Applications

FILES Application

The purpose of the external application **Files** is to allow an import/export of the application configuration data in the form of files in *Microsoft Excel*® format. Once activated, an 'Export' button allows you to obtain the current configuration in the form of an immediately downloadable file. This file includes the following tabs:

Tab	Content
LocationZone	Facility Zones Hierarchy (See <i>Zones</i> en page 88)
LocationZoneType	Zone types (See <i>Zone Types</i> en page 95)
PositionType	Position Types within the zones (See <i>Position Types</i> on page 96)
LocationUnit	Departments/Units
NurseCallDevice	Nurse Call Devices
DriverInstance	Driver Instances
NurseCallMedaillon	Medallions
Patient	Patients
Variable	Variables
AlarmEventGenerator	Alarm Generators
Media	Media
Staff	Employees
Beacon – Sharp	Localization beacons (See <i>Localization Beacons</i> on page 95)
VisitReminderPolicy	Visit reminder policies (See <i>Visit Reminders</i> on page 145)
MediaType	Media types
EventType	Event Type Hierarchy (See <i>Event Types Hierarchy</i> on page 128)

Each tab has specific columns with a significant header, as well as a unique identifier generated during the export.

It is then possible to modify the data in this file outside Ofelia® and then use the 'Import' button to replace the current configuration with that of the file. Keeping the identifiers in the first column allows you to make changes to existing elements, while adding new identifiers allows you to create new elements.

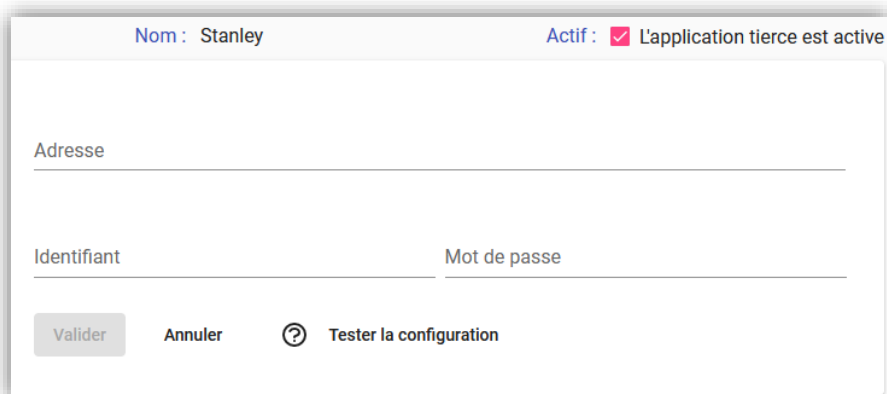
An additional option allows the configuration to be synchronized automatically and regularly at the specified frequency. The data is then read from all the files in *Microsoft Excel*® format present in the [ProgramData\Import] directory and imported into the software.

STANLEY® Application

This application allows you to connect to an existing Stanley® tracking system and import the zone structure of the facility, then each time an alarm is emitted by one of the networked devices (DECT mobiles, smartphones, etc.) the system can obtain their location.

The zone hierarchy of the establishment is recreated in Ofelia® with specific zone types.

The configuration consists in indicating the address of the Stanley® server as well as the authentication parameters with this server (login and password).



The image shows a configuration window for the Stanley application. At the top, it displays 'Nom : Stanley' and 'Actif : ☒ L'application tierce est active'. Below this, there is a large text input field labeled 'Adresse'. Underneath the address field, there are two smaller text input fields: 'Identifiant' on the left and 'Mot de passe' on the right. At the bottom of the form, there are four buttons: 'Valider' (disabled), 'Annuler', a help icon (question mark in a circle), and 'Tester la configuration'.

TELECARE® IP Application

The teleCARE® IP protocol is an IP-based nurse call system. The teleCARE® IP system is designed for hospitals, healthcare facilities, Senior facilities, etc. where it is used to broadcast information generated in the nurse call system, such as calls, nurse presence, medical alarms, technical alarms, etc.

The associated external application also allows the transfer of the facility configuration into Ofelia®, which has already been done during the configuration of the nurse call system. The implementation of the external application essentially consists of defining the address of the NISM central unit, then indicating the correspondence between the types of zones and position of the central unit and their equivalent in Ofelia®:

Détails de l'application tierce 'teleCARE IP'

Nom : teleCARE IP Actif : ☒ L'application tierce est active

Instance de driver NISM2	Adresse 192.168.1.171
-----------------------------	--------------------------

Types de zone

Type de zone pour les communs Espace Commun	Type de zone pour les services Service	Type de zone pour les bâtiments Bâtiment
Type de zone pour les étages Etage	Type de zone pour les couloirs Couloir	Type de zone pour les sanitaires co... Sanitaire commun
Type de zone pour les chambres Chambre	☐ Mettre à jour l'arborescence des zones ☒ Importer les services	

Types de position pour la chambre

Type de position pour le lit Lit	Type de position pour la porte Porte	Type de position pour les toilettes Toilettes
-------------------------------------	---	--

- Driver Instance** in charge of communicating with the remote NISM unit
- IPv4 Address** of the remote unit
- Zone Type**: selection, for each zone type declared on the teleCARE® side, of its equivalent in Ofelia®. Ofelia types must first be created in the system (see *Zone Types* on page 95)
- Update zone tree**: if this box is checked, the zones in Ofelia® will be synchronized to reflect those in the NISM central unit

- 5 **Import departments:** if this box is checked, the departments/units declared in the central unit will be created in Ofelia and the zones associated with their unit
- 6 **Position Types:** mapping between the position types in the rooms defined in the central office and those of Ofelia®. The position types must first be created in the software (see *Position Types* on page 96)

A dedicated button lets the administrator initiate a manual synchronization.

LDAP Directory Application

The LDAP Directory application allows an administrator to configure user authentication through an external directory. It can work with a *Microsoft Active Directory*®, and in a future version, with an LDAP directory (Open LDAP for example). This last option is not active in the current version of Ofelia®, which for the moment checks the membership to a *Microsoft Windows*® domain in order to activate the external application.

Détails de l'application tierce 'LDAPDIRECTORY'

Nom : LDAPDIRECTORY Actif : ☒ L'application tierce est active

Domaine 1 ofelia.fr Délai d'attente en secondes 20 2

☒ Utiliser Kerberos si possible 3

☒ Utiliser SSL 4

☒ Ne pas synchroniser les utilisateurs 5

Groupes de sécurité autorisés 6

Unité organisationnelle 7 Rôle par défaut Exploitation 8

Serveur 9 Port 636 10

Identifiant Mot de passe 11

- 1 **Domain:** Identifier of the remote *Microsoft Active Directory*® forest

- 2 **Timeout in seconds** when trying to reach a domain controller or directory server before considering the connection failed
- 3 **Use Kerberos if available:** If this box is checked, the Kerberos protocol will be used in the first instance before attempting an LDAP connection. This setting is recommended for Single Sign On (SSO) operation with a *Microsoft Active Directory*® domain.
- 4 **Use SSL:** If this box is checked the exchanges with the LDAP server will be encrypted (LDAPS).
- 5 **Do not synchronize users:** by default, the application creates users in the Ofelia® directory automatically when they connect. By checking this box, it is possible to create users in Ofelia® beforehand and then associate them manually with an existing identifier in the *Microsoft Active Directory*®.
- 6 **Authorized security groups:** Multiple choice list of security groups in the domain. Only users belonging to these groups will be allowed to connect and will be synchronized in the Ofelia® directory. This allows you to authenticate with an external directory without authorizing all the users of the domain to access the software.
- 7 **Organizational Unit:** If an Organizational Unit of the domain is selected, only the users belonging to this OU will be synchronized and will be able to connect. This allows authentication with an external directory without allowing all users in the domain to access the software.
- 8 **Default Role:** To be able to use the software, it is necessary to have at least one role (see *Roles* on page 53). During the synchronization, the users automatically created in Ofelia® will have the role indicated here. The administrator will then be able to modify this role to assign another one: it is therefore recommended to choose a role with few privileges here.

The following elements are inactive for the moment, they concern the implementation of a connection to an LDAP directory other than *Microsoft Active Directory*® and will be used in a later version:

- 9 **Server:** LDAP server address (IPv4 or FQDN)
- 10 **Port:** TCP port used to connect to LDAP server (by default, 636 for AD over SSL). Ports #3268 (*Global Catalog*) and 389 (AD) should be avoided since they are usually not encrypted.
- 11 **Login credentials** for the LDAP server.

Notes:

1. For Kerberos operation, the service account running Ofelia® must be allowed to browse the *Microsoft Active Directory*®.
2. By default, the Kerberos mode setting on the external application will result in the implementation of the automatic unified login (SSO) within Ofelia®. In this configuration, a user who has a valid Windows session on the domain and who belongs to the Windows groups and to the organizational unit indicated will be automatically logged in the application, avoiding the need to enter his credentials. The password is not stored in Ofelia®, but authentication is simply delegated to the remote *Microsoft Active Directory*®. It is possible to override this mode by adding the "SsoEnabled"=false key in the appSettingsCustom.json configuration file.