



**STEINEL LIVELINK**

**MQTT INTERFACE**

# CONTENT

---

<b>1. INTRODUCTION</b>	Page 2
<b>2. CONFIGURING THE MQTT INTERFACE IN THE STEINEL LIVELINK APP</b>	Page 2
<b>3. TOPIC OF THE DATA PACKETS</b>	Page 2
<b>4. LIVE DATA OF LUMINAIRES AND SENSORS</b>	Page 3
<b>5. GROUP CONFIGURATIONS</b>	Page 4
<b>6. REQUEST FOR GROUP CONFIGURATIONS</b>	Page 7
<b>7. BUILDING PLAN</b>	Page 8
<b>8. SYSTEM BACKUP</b>	Page 11
<b>9. TEST RESULTS OF THE DALI EMERGENCY LIGHTING</b>	Page 11
<b>10. SYSTEM EVENTS</b>	Page 12

# 1. INTRODUCTION

Modern DALI lighting systems not only enable intelligent control possibilities, but also provide a wide range of luminaire and sensor data.

Using the LiveLink control system, this data can be read out from a lighting installation and made available via an MQTT interface. The interface is configured using the STEINEL LIVELINK app.

The following data and configurations are provided either on request or cyclically:

- Live data of luminaires and sensors
- Configuration of luminaire groups and associated light scenes
- Building plans
- System backups
- Test results of DALI emergency lighting
- System-relevant information

The data packets are transmitted in JSON format.

## 2. CONFIGURING THE MQTT INTERFACE IN THE STEINEL LIVELINK APP

The MQTT interface is configured in the STEINEL LIVELINK app. To do this, the IP address and port for receiving the data and the authentication settings (user name and password or certificates used for authentication) are stored.

The transmission interval (from one to four minutes) for the live data and compression (if required) can be selected.

## 3. TOPIC OF THE DATA PACKETS

The single data packets are stored in the following topics on the target server:

Live data	11/{UserName}/data/live	Cyclical transmission of luminaire and sensor data at set intervals.
Group configuration	11/{UserName}/data/groups	Transmitted when the configuration changes and after a restart of the system.
Building plans	11/{UserName}/data/floorplan	Transmitted when the configuration changes.
System backup	11/{UserName}/data/archive/{backupName}	Transmitted after a backup has been created.
Test results of the DALI emergency lighting	11/{UserName}/data/emergency/protocols/{testProtocolName}	Test results (function- and duration tests) of the emergency lighting are transmitted automatically.
System events	11/{UserName}/data/events	Transmitted when a system event occurs.

## 4. LIVE DATA OF LUMINAIRES AND SENSORS

Field	Type	Description	
timestamp	Integer	Time at which live data was transmitted, expressed as UNIX timestamp (number of seconds elapsed since 00:00 UTC on Thursday, 1 January 1970).	
values	Object[]	List of all connected luminaires and their data.	
(Number of luminaires)	id	String	Unique ID of the luminaire.
	update Interval	Integer	Set update interval in seconds.
	level	Integer	DALI level of the luminaire (between 0 and 254).
	on	Boolean	Switching status of the luminaire (on/off).
	operating Time	Integer	Total operating time of the luminaire whilst switched on in minutes.
	perceivedBrightness	Integer	Perceived brightness of the luminaire as a percentage based on the actual DALI value.
	power	Object[]	Power consumption of the luminaire.
	current	Number	Momentary power consumption of the luminaire.
	minimum	Number	Lowest power consumption of the luminaire over the entire operational period.
	maximum	Number	Highest power consumption of the luminaire over the entire operational period.
	switchCount	Integer	Number of switching cycles.
	temperature	Object[]	Temperature in the luminaire (degrees Celsius).
	current	Number	Momentary measured temperature in LED driver of the luminaire.
	minimum	Number	Lowest measured temperature in LED driver of the luminaire over the entire operational period.
	maximum	Number	Highest measured temperature in LED driver of the luminaire over the entire operational period.
	emergency	Object[]	Information from the emergency luminaire if it is a DALI DT1 emergency luminaire (in accordance with DIN EN 62386-202:2010).

Field	Type	Description	
mode	String	Current operating mode of the emergency luminaire. Bit 0: rest mode, Bit 1: normal mode, Bit 2: emergency mode, Bit 3: extended emergency mode, Bit 4: function test active, Bit 5: duration test active, Bit 6: inhibiting input active, Bit 7: switch input is on.	
inhibitingInputActive	Boolean	Is the lock input of the emergency luminaire activated? (True: Yes / False: No).	
switchInputOn	Boolean	Status of the emergency luminaire switch input (True: active / False: inactive).	
failureStatus	String	Failure status of the emergency luminaire. Bit 0: circuit defect, Bit 1: rated duration of the battery insufficient, Bit 2: battery defect, Bit 3: emergency lamp defect, Bit 4: maximum test execution time-out exceeded for function test, Bit 5: maximum test execution timeout exceeded for duration test, Bit 6: function test failed, Bit 7: duration test failed.	
batteryCharge	Number	Battery charge, 0 = 0%, 254 = 100%, 255 = unknown (e.g. no battery connected).	
sensorValues (No. of luminaires)	Object[]	List of all connected DALI sensors and their measured values.	
	id	String	Unique ID of each sensor.
	measureValue	Number	Measured value of each sensor.

## 5. GROUP CONFIGURATIONS

The group configuration provides an overview of the luminaires in the group and the sensors linked to the group. All scenes are also specified in which the group is included, and the group's brightness and colour temperature settings. The group configuration is transmitted whenever a configuration change is made or the system is restarted.

### Group configuration data packet

Field	Type	Description
id	String	Unique ID of the group: e.g. "47d30a82-0081-4403-a074-5a56d05bcb75".
name	String	Name of the group (only available if set).
actionCount	Number	Number of functions supported by the devices in the group.

Field	Type	Description
(0 ... actionCount) actions	Object[]	List of all functions supported by this group (e.g. switching, dimming, colour temperature or emergency light) [0 ... actionCount].
	String	Group function; possible types include Identify, OnOff, LevelControl, ColorControl, FunctionTest, DurationTest, CustomAction, UnknownAction.
	String	URL for access via REST – no relevant information
deviceComponent	Boolean	Indicates whether the group consists of physically present components (luminaires or sensors), or whether it is a virtual or logical group. (True: Yes / False: No).
active	Boolean	The group is active and is monitored by the LiveLink control (True: Yes / False: No).
deviceCount	Number	Number of devices/luminaires in the group
(0 ... deviceCount) devices	Object[]	List of all associated devices [0 ... deviceCount] and their properties.
	String	Unique ID of each device.
	String	Name of each device.
	String	Device type: dimmablelight, dt8light, emergencylight, identifiablelightsensor, identifiablemotionsensor, identifiableswitch, onofflight, rgblight, rgbwlight, simplelightsensor, simplemotionsensor, simpleswitch, wwilight
	String	URL for access via REST – no relevant information
sensorCount	Number	Number of sensors belonging to this group.
(0 ... sensorCount) sensors	Object[]	List of all associated sensors [0 ... sensorCount] and their properties and measured values.
	String	Unique ID of each sensor (also used in live data as sensorValues[].id).
	String	Display name of each sensor.
	String	Sensor type: identifiablelightsensor, identifiablemotionsensor, simplelightsensor, simplemotionsensor

Field	Type	Description	
url	String	REST path for the sensor – no relevant information	
measurement	String	Contains the physical parameter of the sensor, e.g. brightness, temperature or humidity.	
unit	String	Unit of measurement of the measured value, e.g. lux or °C.	
minimum	Number	Minimum measured value that is transmitted with the live data.	
maximum	Number	Maximum measured value that is transmitted with the live data.	
eventSupport		Can be ignored.	
active	Boolean	Can be ignored.	
sceneCount	Number	Number of scenes in which this lighting group is included.	
scenes	Object[]	List of all associated scenes [0 ... SceneCount] and their properties.	
(0 ... SceneCount)	id	String	Unique ID of each scene.
	name	String	Name of each scene.
	url	String	URL for access via REST – no relevant information
	sceneMemory	Object[]	Specifies the scene memory. This contains all action parameters that can be edited.
	onoff	Object[]	Indicates the on/off status of the luminaire group.
	on	Boolean	Indicates whether the luminaire group is switched on (True: On / False: Off).
	levelcontrol	Object[]	Indicates the brightness setting of the luminaire group.
	minLevel	Number	Indicates the minimum luminous flux of the group (as a DALI value).
	maxLevel	Number	Indicates the maximum luminous flux of the group (as a DALI value).
	level	Number	Indicates the luminous flux of the group in this scene (from 1% to 100%).

Field	Type	Description
perceivedBrightness	Number	Indicates the perceived brightness of the group in this scene.
colorcontrol	Object[]	Indicates the colour temperature properties of the luminaire group (optional, only for groups with colour temperature-capable luminaires).
colorTemperature	Number	Indicates the colour temperature of the scene in Kelvin.
hue	Number	Indicates the colour hue of the luminaire group in the scene.
saturation	Number	Indicates the level of colour saturation of the luminaire group in the scene.
statistic	Object[]	Indicates whether the group supports statistical data.
active	Boolean	Indicates whether the group supports statistical data (True: Yes / False: No).
type	String	Group type: onoffgroup, dimmablegroup, colorgroup
visible	Boolean	Can be ignored.

## 6. REQUEST FOR GROUP CONFIGURATIONS

By executing the request `"/system/livelinkconfiguration"`, the group configuration can be called, also during operation. This is transmitted to the topic `"ll/{UserName}/data/groups"`.

Request	<code>ll/{UserName}/request/{requestId}/system/livelink-configuration</code>	Sent when the configuration changes and after a restart of the system.
Response	<code>ll/{UserName}/request/{requestId}/code/{responseStatusCode}</code>	Response with status code: <ul style="list-style-type: none"> <li>• 200: OK, group configurations will be newly transmitted</li> <li>• 400: invalid command or invalid RequestId</li> </ul>

The RequestId is a unique UUID that identifies only the request. This is usually generated by the calling service (e.g.: `"a4144619-0a2f-43c4-8a32-4b735033b428"`).

## 7. BUILDING PLAN

After every configuration change the building plan is transmitted, including all room information (room name and dimensions) and the positions of all luminaires and sensors on the plan. The building plan is transmitted in compressed gzip format.

### Building plan data packet

Field	Type	Description	
projectsCount	Number	Number of projects in the system, with each floor representing a project.	
projects	Object[]	List of all projects or floors [0 ... projectsCount] and their properties.	
(0 ... projectsCount)	projectId	Number	Project ID
	projectName	String	Project name
	description	String	Description of the project
	author	String	Project author
	roomCount	Number	Number of rooms in the project or on the floor.
	rooms	Object[]	List of all rooms [0 ... roomCount] and their properties and components.
(0 ... roomCount)	roomId	Number	Room ID
	specificRoomId	String	Specific room designation.
	name	String	Room name
	description	String	Description of the room.
	floor	Number	ID of the floor to which the room belongs.
	geometry	Object[]	Geometry of the room.
	baseArea	Object[]	Specifies a starting point for the room and determines where on the building plan the room is fundamentally located.
	x	Number	Horizontal position of the room.
	y	Number	Vertical position of the room.
	offset	Object[]	The offset shifts the room in addition to the starting point. It is used to precisely position rooms without changing their shape.
x	Number	Horizontal shift	

Field	Type	Description
y	Number	Vertical shift
roomGeometry	Object[]	Indicates the shape of the room. The room is specified as a line consisting of several points. Each point is a corner of the room. The points are connected in sequence to form the outline of the room.
x	Number	Horizontal position of the corner.
y	Number	Vertical position of the corner.
lightCount	Number	Number of luminaires in the room.
lights	Object[]	List of all luminaires [0 ... lightCount] and their properties.
positionId	Number	Can be ignored.
model	String	Can be ignored.
manufacturer	String	Can be ignored.
type	String	Can be ignored "luminaire"
toc	String	Can be ignored.
position	Object[]	Indicates the spatial position of the luminaire in the room.
x	Number	X coordinate of the luminaire in the room.
y	Number	Y coordinate of the luminaire in the room.
z	Number	Z coordinate of the luminaire in the room.
technology	Object[]	Mapping of the luminaires to the LiveLink configuration.
uuid	String	UUID of the mapped luminaire.
uinterfaceld	String	MAC address of the gateway and DALI line to which the luminaire is connected.
sensorCount	Number	Indicates how many sensors are in this room.
sensors	Object[]	List of all sensors spatially mapped to this room [0 ... sensorCount] and their properties.
positionId	Number	Position ID of the sensor in the corridor plan (unique in the building plan).

(number of corners)

(0 ... lightCount)

(0 ... sensorCount)

Field	Type	Description
model	String	Model name of the sensor.
manufacturer	String	Manufacturer of the sensor.
type	String	Object type in the building plan.
toc	String	Additional/classification field from the export (may be empty).
position	Object[]	Position of sensor in the spatial coordinate system.
x	Number	X coordinate (relative to the room origin).
y	Number	Y coordinate (relative to the room origin).
z	Number	Installation height above the floor.
technology	Object[]	Mapping of the sensor to the LiveLink configuration.
uuid	String	UUID of the mapped sensor.
interfaceld	String	MAC address of the gateway and DALI line to which the sensor is connected.
switchCount	Number	Push-button coupler
switches	Object[]	List of all switches spatially mapped to this room [0 ... switchCount] (currently not available).
positionId	Number	Internal position of the switch on the building plan.
model	String	Model name of the switch.
manufacturer	String	Manufacturer of the switch.
type	String	Object type in the building plan.
toc	String	Additional/classification field from the export (may be empty).
position	Object[]	Position of switch in the spatial coordinate system.
x	Number	X coordinate (relative to the room origin).
y	Number	Y coordinate (relative to the room origin).
z	Number	Installation height above the floor.
technology	Object	Mapping of the switch to the LiveLink configuration.

(0 ... switchCount)

Field	Type	Description
uuid	String	Unique UUID of the switch for technical mapping (if listed as a device in the system).
interfaceld	String	MAC address of the gateway and DALI line to which the switch is connected.
floors	Object[]	Indicates the floors.
identifier	Number	Floor identifier, reference to the building structure
displayName	String	Name of the floor.
backgroundImage	String	Can be ignored.

## 8. SYSTEM BACKUP

As soon as a new backup has been created in the LiveLink system it is transmitted via MQTT. The backup is transmitted as a ".tar.gz" archive containing the complete configuration of the LiveLink system. The name of the backup consists of the prefix "backup", the date and the local time of the LiveLink system (e.g.: "backup\_2025\_08\_29\_10\_52.tar.gz"). With an automatic backup the prefix is "autobackup".

## 9. TEST RESULTS OF THE DALI EMERGENCY LIGHTING

For every emergency luminaire that has undergone an automated test (function- or duration test), the test result is reported (in accordance with DIN EN 62386-202:2010).

### Data packet with the test reports

Field	Type	Description
testType	String	Type of test ("function test" or "duration test").
id	String	Unique ID of the emergency luminaire.
name	String	Name of the emergency luminaire.
timestamp	Integer	Time of data transmission, expressed as UNIX timestamp (number of seconds elapsed since 00:00 UTC on Thursday, 1 January 1970). Corresponds to the date of the test.
gtin	String	GTIN (Global Trade Item Number) of the emergency luminaire. Transmitted in Regex format: "^0x([a-fA-F0-9]{12})\$".
serialNumber	String	Serial number of the emergency luminaire, transmitted in Regex format: "^0x(([a-fA-F0-9]{8}) ([a-fA-F0-9]{16}))\$".

Field	Type	Description
failureStatus	String	<p>Failure status of the emergency luminaire, transmitted in Regex format: “^0x([a-fA-F0-9]{2})\$”.</p> <p>The following status messages may occur: Bit 0: circuit defect, Bit 1: rated duration of the battery insufficient, Bit 2: battery defect, Bit 3: emergency lamp defect, Bit 4: maximum test execution time-out exceeded for function test, Bit 5: maximum test execution timeout exceeded for duration test, Bit 6: function test failed, Bit 7: duration test failed.</p> <p>If there is no failure status, the result “0x00” is transmitted.</p>
emergencyStatus	String	<p>Emergency status of the emergency lighting component, transmitted in Regex format: “^0x([a-fA-F0-9]{2})\$”.</p> <p>The following emergency status messages may occur:            Bit 0: lock mode, Bit 1: function test finished and result valid, Bit 2: duration test finished and result valid, Bit 3: battery fully charged, Bit 4: function test pending, Bit 5: duration test pending, Bit 6: identification active, Bit 7: physical selected.</p>
ratedDuration	Integer	Nominal operating time for the emergency luminaire duration test.
durationTestResult	Integer	Result of the duration test in minutes. Must be greater than or equal to the nominal operating time.
testPassed	Boolean	Returns the result of the test (True: test passed / False: test not passed).

## 10. SYSTEM EVENTS

As soon as an event occurs in a LiveLink system, it is immediately transmitted. This can be either an information or an error. The following events can occur in the LiveLink system:

Code	Short text	Meaning	Event type
<b>System Events</b>			
0x0001'0000	CoreStarted	LiveLink system is starting up	Info
0x0001'0001	CoreStopped	LiveLink system has been paused	Info

Code	Short text	Meaning	Event type
0x0002'0000	DirtyAppRestart	System restart	Error
0x0003'0000	SessionFileConvertError	Session file conversion error	Error
0x000f'ffff	UnknownError	An unknown error has occurred	N/A

#### DALI Events

0x0200'0000	ECG_DISCONNECT_DETECTED	Control gear unit has no connection / control gear unit is connected	Error (on set) / Info (on reset)
0x0200'0002	ECG_LAMP_FAILURE_DETECTED	Luminaire error has occurred / luminaire error has been rectified	Error (on set) / Info (on reset)
0x0200'0010	ECG_EMERGENCY_LIGHT_FAILURE_DETECTED	Emergency lighting error has occurred / emergency lighting error has been rectified	Error (on set) / Info (on reset)
0x0200'0012	ECG_EMERGENCY_LIGHT_FUNCTION_TEST_DEACTIVATION_DETECTED	Automatic function test deactivated / automatic function test reactivated	Error (on set) / Info (on reset)
0x0200'0014	ECG_EMERGENCY_LIGHT_DURATION_TEST_DEACTIVATION_DETECTED	Automatic duration test deactivated / automatic duration test deactivated	Error (on set) / Info (on reset)
0x0201'0000	SENSOR_DISCONNECT_DETECTED	Sensor is not connected / sensor is connected	Error (on set) / Info (on reset)
0x0202'0000	SWITCH_DISCONNECT_DETECTED	Switch is not connected / switch is connected	Error (on set) / Info (on reset)
0x020f'0000	INTERFACE_DISCONNECT_DETECTED	DALI gateway has no connection / DALI gateway is connected	Error (on set) / Info (on reset)

#### KNX Events

0x0401'0000	BaosOffline	KNX gateway is offline / KNX gateway is online	Error (on set) / Info (on reset)
0x0402'0000	KnxOffline	KNX Online / KNX Offline	Error (on set) / Info (on reset)

<b>Code</b>	<b>Short text</b>	<b>Meaning</b>	<b>Event type</b>
<b>Network Events</b>			
0x0801'0000	ClientDisconnected	Device disconnected / device connected	Error (on set) / Info (on reset)
0x0801'0002	CollisionDetected	Network error has occurred	Info
0x0801'0003	CollisionCouldNotBeAvoided	A network error cannot be avoided	Info
0x0801'0020	WiFiShutdownWiFiOff	Wi-Fi switched off	Info
0x0801'0021	WiFiShutdownWiFiOnAgain	Wi-Fi has been switched on	N/A
0x0801'0100	WiFiModuleCrash	Wi-Fi module driver has crashed	Error
<b>Emergency Lighting Events</b>			
0x0100'0000	NoError	No error has occurred	Info
0x0100'0001	ModbusConnectionFailed	Connection to the bus could not be made / connection to the bus made	Error (on set) / Info (on reset)
0x0100'0002	ModbusReadRegistersFailed	Error occurred while reading the data	Error

**STEINEL GmbH**

Dieselstraße 80-84  
D-33442 Herzebrock-Clarholz  
+49 (0) 5245/4 48-0  
info@steinel.de  
www.steinel.de