Dashboard DB Tables

Config_dashboard

Each row of this table represents the details and configuration parameters of a **single dashboard** created and instantiated in the Dashboard Manager.

Config_dashboard				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
Id	INT(11)	Auto-incremental integer ID.	PRIMARY	
name_dashboard	VARCHAR(300)	Name of the dashboard (as a unique identifier).		
title_header	VARCHAR(300)	Title of the current dashboard to be displayed in header.		
subtitle_header	VARCHAR(300)	Sub-title of the current dashboard to be displayed in the header.		
color_header	VARCHAR(40)	Color of the current dashboard header.		
width	INT(11)	Width in pixels of the current dashboard.		
height	INT(11)	Height in pixels of the current dashboard.		
num_rows	INT(11)	Number of rows of the current dashboard (**DEPRECATED**).		
num_columns	INT(11)	Number of columns of the current dashboard.		
user	VARCHAR(100)	Name of the user who created the current dashboard.		ActuatorsAppsValues.username / NO ActuatorsEntitiesValues.username / NO Config_widget_dashboard.creator / NO IotApplications.user / NO NodeRedInputs.user / NO NodeRedMetrics.user / NO OperatorEvents.user / NO Users.username / NO UsersPoolsRelations.username / NO
status_dashboard	INT(11)	Status of the current dashboard. It can assume the following values: 0 ➔ inactive, 1 ➔ active.		
creation_date	TIMESTAMP	DateTime when the current dashboard has been created.		
color_background	VARCHAR(40)	Background color of the current dashboard.		
external_frame_color	VARCHAR(40)	Color of the external frame of the dashboard, whith respect to the widget area.		
headerFontColor	VARCHAR(40)	Font color of the dashboard header.		
headerFontSize	INT(3)	Font size of the dashboard header.		
logoFilename	VARCHAR(45)	File Name of the logo, if present, to be displayed on the right side of the dashboard header.		

logoLink	VARCHAR(400)	Web link to be opened on click on the logo represented by *Config_dashboard.logoFileName*.		
visibility	VARCHAR(45)	This field specifies if the current dashboard is visible by all users (“public”) or only by the user who created it (“author”).		
headerVisible	INT(1)	Integer values specifying if the current dashboard’s header is visible (1) or not visible (0).		
embeddable	VARCHAR(3)	This field indicates if the current dashboard can be embedded (“yes”) or not (“no”) in an iFrame for external use/contexts.		
authorizedPagesJson	TEXT	This field contains a JSON with a list of authorized web pages which can host and display the current widget (through the iFrame created with the “Embedding” button in the dashboard menu). For other web pages not present in this list, the embedded frame is automatically hidden.		
viewMode	VARCHAR(45)	Dashboard visualization modality. It can be fixed or horizontally responsive. Expected values: {alwaysResponsive; fixed; largeResponsive; mediumResponsive}.		
fromNodeRed	VARCHAR(3)	This field indicates if the current dashboard has been created within a Node-RED application (“yes”) or not (“no”).		
gridColor	VARCHAR(40)	Color of the background grid used in Edit Mode.		
nrGpsRelativeUrl	TEXT	**DEPRECATED**		
last_edit_date	TIMESTAMP	DateTime (YYYY:MM:DD HH:mm:ss) of the last edit made on current dashboard.		
lastUsedColors	TEXT	Last colors used in the color palette of the current dashboard.		
Deleted	VARCHAR(3)	It indicates if the current dashboard has been deleted (“yes”) or not (“no”).		
screenshotFilename	VARCHAR(150)	File name of the screenshot generated for the current dashboard.		
bckImgFilename	VARCHAR(150)	File name of the image used as background in the current dashboard.		
useBckImg	VARCHAR(3)	This field indicates if the current dashboard actually uses an image (represented by *Config_dashboard.bckImgFilename*) as background (“yes”) or not (“no”).		
backOverlayOpacity	FLOAT			

Config_widget_dashboard

Each row of this table represents configuration parameters of a **single instantiated widget**, showing also the relationship to its corresponding dashboard (i.e.: the dashboard in which the widget has been instantiated).

Config_widget_dashboard				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
Id	int(11)	Auto-incremental integer ID.	PRIMARY	
name_w	varchar(150)	Name (Internal Unique Identifier) of the current widget.		
id_dashboard	Int(11)	Id of the dashboard containing the current widget.	Config_widget_dashboard_ibfk_1	Config_dashboard.Id / YES DashboardsViewPermissions.IdDashboard / NO IdDashDailyAccess.IdDashboard / NO
id_metric	varchar(150)	Id of the metric observed/measured by the current widget.		
type_w	varchar(150)	Type of the current widget (e.g.: “widgetBarContent”, “widgetGaugeChart”, “widgetTimeTrend”, “widgetSpeedometer” etc.).		Widgets.id_type_widget / NO WidgetsIconsMap.mainWidget / NO
n_row	int(11)	Position on the Y-Axis for the current widget in the dashboard (considering that the origin (0, 0) of the dashboard is placed at the upper-left corner of the visualization area).		
n_column	int(11)	Position on the X-Axis for the current widget in the dashboard (considering that the origin (0, 0) of the dashboard is placed at the upper-left corner of the visualization area).		
size_rows	int(11)	Vertical dimension (expressed in number of rows) of the current widget.		
size_columns	int(11)	Horizontal dimension (expressed in number of columns) of the current widget.		
color_w	varchar(40)	Default color for the current widget.		
frequency_w	varchar(100)	Frequency of refresh (expressed in number of seconds) of the current widget value. It is displayed in the upper-right corner of the widget, next to the contextual menu control.		
temporal_range_w	varchar(100)	This field is used for time-trend and time-trend-compare widgets: it specifies the temporal range for the displayed time series.		
municipality_w	varchar(100)	Some widgets only (e.g. Weather Forecast, First Aid, Protezione Civile) require the municipality information to query the Km4City Smart City API to add enriched data. In these cases, the municipality information is stored in this field.		

infoMessage_w	text	Additional textual information for the current widget. It is displayed in a dedicated modal by pushing the "i" informative button in the upper-left corner of the widget.		
link_w	varchar(1024)	Web link to be opened by directly clicking upon the widget.		
parameters	text	JSON containing functional and style parameters for the current widget.		
udm	varchar(45)	Data Type (i.e.: measure unit) of the measure observed by the current widget (used only for Single Content widgets).		
udmPos	varchar(50)	Default position of the *udm* (Data Type) field above described, within the widget body.		
fontSize	int(3)	Size of the font used in the current widget.		
fontColor	varchar(40)	Color of the font used in the current widget. **It's going to be DEPRECATED** actually this information is now also contained in the *styleParameters* field.		
frame_color_w	varchar(40)	Color of the frame in which the current widget is contained when it is instantiated. **It's going to be DEPRECATED** actually this information is now also contained in the *Config_widget_dashboard.styleParameters* field.		
controlsPosition	varchar(25)	Default position where the zoom controls for the current widget (iFrame only) are placed within the widget body, in Edit mode only. **DEPRECATED** (it's not shown anymore).		
showTitle	varchar(3)	This field specifies if the title of the current widget has to be shown or not. Expected values: {"yes"; "no"}.		
controlsVisibility	varchar(15)	This field specifies the visualization modality of the zoom controls for the current widget (iFrame only), e.g.: "alwaysVisible", "hidden". **DEPRECATED** (it's not shown anymore).		
zoomFactor	float	Float value representing the zoom factor (1=100%) for displaying the current widget.		
defaultTab	int(2)	Used only for those widgets which have sliding tabs (Twitter quotations & mentions, Protezione Civile). Expected values: {-1, 0, 1, …, *n*}, where: -1 → automatic sliding of all tabs; 0 → 1st tab as default visualization; 1 → 2nd tab as default visualization; … *n* → *(n-1)* tab as default visualization.		
zoomControlsColor	varchar(40)	**DEPRECATED**.		
scaleX	double	Zoom visualization scale on x-axis for the current widget. **DEPRECATED**		
scaleY	double	Zoom visualization scale on y-axis for the current widget. **DEPRECATED.**		
headerFontColor	varchar(40)	Color of the font used in the header of the current widget.		

styleParameters	text	Some widgets usually have complex style with lots of parameters. Such information for the current widget is included in a JSON which is stored in this field.		
infoJson	text	JSON containing user defined values for labelling time series (i.e.: time series graph labels legend) in time-series based widgets.		
serviceUri	varchar(600)	This field contains the Km4City Service URI of a Smart City resource, in case this information is needed to query the Km4City Smart City API to provide additional, enriched information.		
viewMode	varchar(50)	This field specifies the visualization modality for the current widget (e.g.: “list”, “singleDetails”, “singleSummary”, “hospitalsOverview”, “map”).		
hospitalList	text	List of Km4City URIs of hospitals visualized on the current widget (for FirstAid widgets only).		
lastSeries	text	This field is used only for widgets based on data series. In this case, this field contains a JSON representing the last data serie (not scalar) displayed for the current widget. When a certain service updates the serie values for the current widget, also the *lastSeries* value is updated.		
notificatorRegistered	varchar(3)	This field specifies whether the current widget has been registered or not on the Notificator to send alerts. This is used for the first registration only (if a widget has been registered, the value of this field is kept to “yes”). Then, the updated status of the current widget in the Notificator (enabled / not enabled) is contained in the next *Config_widget_dashboard.notificatorEnabled* field. Expected values: {“yes”, “no”}.		
notificatorEnabled	varchar(3)	After a widget has been registered for the first time on the Notificator, the user can enable/disable the notification feature. This information is stored in this field. Expected values: {“yes”, “no”}.		
oldParameters	text	**DEPRECATED.**		
enableFullscreenTab	varchar(3)	This field specifies if the current widget can be viewed at full-screen in a new browser tab, by clicking the button  on the upper-right corner of the widget.		
enableFullscreenModal	varchar(3)	This field specifies if the current widget can be viewed at full-screen on the current browser tab in a full-screen modal, by clicking the button  on the upper-right corner of the widget.		
fontFamily	varchar(100)	Font type used in the current widget (used only for Selector and Selector-Web widgets).		

entityJson	text	JSON describing the IOT Broker entity attributes for the current widgets (actually used only for actuator widgets, i.e.: having *Config_widget_dashboard.actuatorTarget* field set to “broker”).		
attributeName	varchar(100)	Name of the attribute, in the IOT broker JSON entity describing the attribute upon which the current widget is acting/reading (now used for actuator-widgets, i.e.: having *Config_widget_dashboard.actuatorTarget* = “broker”).		
creator	varchar(100)	Name of the user who created the current widget.		ActuatorsAppsValues.username / NO ActuatorsEntitiesValues.username / NO Config_dashboard.user / NO IotApplications.user / NO NodeRedInputs.user / NO NodeRedMetrics.user / NO OperatorEvents.user / NO Users.username / NO UsersPoolsRelations.username / NO
lastEditor	varchar(100)	Name of the user who last edited the current widget.		
canceller	varchar(100)	Name of the user (if any) who cancelled the current widget.		
creationDate	varchar(40)	DateTime (YYYY:MM:DD HH:mm:ss) of creation of the current widget.		
lastEditDate	varchar(40)	DateTime (YYYY:MM:DD HH:mm:ss) of last edit of the current widget.		
cancelDate	varchar(40)	DateTime (YYYY:MM:DD HH:mm:ss) of cancellation of the current widget. If the widget has never been cancelled, this fields is set to NULL		
actuatorTarget	varchar(45)	Target entity of the current actuator widget e.g.: “broker”.		
actuatorEntity	varchar(45)	**DEPRECATED.**		
actuatorAttribute	varchar(45)	This field has the same meaning of the *Config_widget_dashboard.attributeName*, but it is referred to Node-RED applications instead of IOT brokers. Therefore, this field specifies the name of the attribute Node-RED application describing the attribute upon which the current widget is acting/reading (now used for actuator-widgets).		
chartColor	varchar(20)	This field specifies the color of chart for graph-based widgets.		
dataLabelsFontSize	int(3)	Font size of data labels, that is the labels of actual values displayed by the current widget. **Not used for now**.		
dataLabelsFontColor	varchar(20)	Font color of data labels, that is the labels of actual values displayed by the current widget. **Not used for now**.		

chartLabelsFontSize	int(3)	Font size of chart labels, that is the labels of the reference system (Cartesian axis or others…) used by the current widget to display actual data values. **Not used for now**.		
chartLabelsFontColor	varchar(20)	Font color of chart labels, that is the labels of the reference system (Cartesian axis or others…) used by the current widget to display actual data values. **Not used for now**.		
appId	varchar(400)	Unique ID of the Node-RED application, if the current widget is associated with a Node-RED app.		NodeRedMetrics.appId / NO NodeRedInputs.appId / NO
flowId	varchar(400)	Unique ID of the Node-RED flow, if the current widget is associated with a Node-RED app (a Node-RED app can contain multiple Node-RED flows).		NodeRedMetrics.flowId / NO NodeRedInputs.flowId / NO
nrMetricType	varchar(400)	Type of metric from / to Node-RED, if the current widget is associated with a Node-RED app.		
nodeId	varchar(400)	Unique ID which is automatically assigned by Node-RED to each Node-RED block.		NodeRedMetrics.nodeId / NO NodeRedInputs.nodeId / NO
sm_based	varchar(30)	This field specifies if the current widget handle data from the Km4City Servicemap Knowledge Base (“yes”) or if the current widget relies on traditional metrics (“no”), or either if it reads Personal Data (“myPersonalData”) by querying the Ownership/Delegation API.		DasbhoardWizard.sm_based / NO
rowParameters	text	This is used only for those widgets which need a parameter to get the needed data. Different cases may occur: - Weather forecast widgets: in this case this field represents the Municipality Name (used to query the Km4City API); - First Aid widgets: in this case this field represents the Municipality Name (used to query the Km4City API); - Widgets reading data from Node-RED personal metrics: in this case this field represents the ID of the generated Node-RED metric (the same as in table *NoderedMetrics.name*); - Widgets using data from Node-RED inputs: in this case this field represents the ID of the generated Node-RED metric (the same as in table *NoderedInputs.name*).		
sm_field	Text	- In case the field *Config_widget_dashboard.sm_based* = “yes” this field specifies the attribute / property to be read from the JSON response of the Km4City API); - In case the field *Config_widget_dashboard.sm_based* = “no” this field is NULL; - In case the field *Config_widget_dashboard.sm_based* = “myPersonalData” this field specifies the attribute /		

		property to be read from the JSON response of the Ownership/Delegation API).		

DashboardTemplates

Each row of this table describes a different dashboard template and its mapping against the *DashboardWizard* categories.

DashboardTemplate				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental id of the current dashboard template.	PRIMARY	
name	varchar(100)	Name of the current dashboard template.		
title	varchar(200)	Title displayed for users of the current dashboard template.		
icon	varchar(200)	File path of the icon used to display the current dashboard template.		
available	varchar(5)	This field specifies if the current dashboard template is available (i.e., it can be used to produce dashboards) or not.		
widgetType	varchar(50)	Type of widget used to instantiate the current dashboard template.		
highLevelTypeSelection	text	This field specifies the High-Level Type value to be filtered in the Dashboard Wizard when the current dashboard template is instantiated. If this field is set to “any”, then no filter is applied in the next step for the Dashboard Wizard.		
natureSelection	text	This field specifies the Nature value to be filtered in the Dashboard Wizard when the current dashboard template is instantiated. If this field is set to “any”, then no filter is applied in the next step for the Dashboard Wizard.		
subnatureSelection	text	This field specifies the Subnature value to be filtered in the Dashboard Wizard when the current dashboard template is instantiated. If this field is set to “any”, then no filter is applied in the next step for the Dashboard Wizard.		
valueTypeSelection	text	This field specifies the Value Type value to be filtered in the Dashboard Wizard when the current dashboard template is instantiated. If this field is set to “any”, then no filter is applied in the next step for the Dashboard Wizard.		
valueNameSelection	text	This field specifies the Value Name value to be filtered in the Dashboard Wizard when the current dashboard template is instantiated. If this field is set to “any”, then no filter is applied in the next step for the Dashboard Wizard.		
dataTypeSelection	text	This field specifies the Data Type value to be filtered in the Dashboard Wizard when the current dashboard template is instantiated. If this field is set to “any”, then no filter is applied in the next step for the Dashboard Wizard.		
highLevelTypeVisible	varchar(5)	This field indicates if the High-Level Type category will be visible in the Dashboard Wizard in the next step for dashboard creation. Expected values: {“true”, “false”}.		

natureVisible	varchar(5)	This field indicates if the Nature category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
subnatureVisible	varchar(5)	This field indicates if the Subnature category will be visible in the Dashboard Wizard in the next step for dashboard creation. Expected values: {“true”, “false”}.		
valueTypeVisible	varchar(5)	This field indicates if the Value Type category will be as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
valueNameVisible	varchar(5)	This field indicates if the Value Name category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
dataTypeVisible	varchar(5)	This field indicates if the Data Type category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
lastDateVisible	varchar(5)	This field indicates if the Last Date category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
lastValueVisible	varchar(5)	This field indicates if the Last Value category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
healthinessVisible	varchar(5)	This field indicates if the Healthiness category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
lastCheckVisible	varchar(5)	This field indicates if the Last Check category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		
ownershipVisible	varchar(5)	This field indicates if the Ownership category will be visible as a column of the Dashboard Wizard table in the next step for dashboard creation. Expected values: {“true”, “false”}.		

DashboardWizard

Each row of this table…

DashboardWizard				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***

id	int(11)	Auto-incremental id of the current dashboard template.	PRIMARY	
nature	varchar(128)	Categorization by NATURE (2nd level) of ingested data/metric to be displayed in the Dashboard Wizard for creating different type of widgets from data.		
high_level_type	varchar(128)	Categorization by HIGH LEVEL TYPE (1st level) of ingested data/metric to be displayed in the Dashboard Wizard for creating different type of widgets from data.	uniqueKey	
sub_nature	varchar(128)	Categorization by SUB NATURE (3rd level) of ingested data/metric to be displayed in the Dashboard Wizard for creating different type of widgets from data.	uniqueKey	
low_level_type	varchar(128)	Categorization by LOW LEVEL TYPE (4th level) of ingested data/metric to be displayed in the Dashboard Wizard for creating different type of widgets from data.	uniqueKey	
unique_name_id	varchar(128)	Unique identifier (name) of each ingested data in the Wizard.	uniqueKey	
Instance_uri	varchar(256)	Unique URI of each ingested data in the Wizard.	uniqueKey	
get_instances	varchar(128)	Method, primitive etc. to retrieve the instance of each ingested data (can be a parameter, a URL for http API call etc.)	uniqueKey	
last_date	datetime	Date of the last ingested value for the specific datum/metric.		
last_value	varchar(128)	Last ingested value for the specific datum/metric.		
unit	varchar(128)	Measure unit of the specific datum/metric.		
metric	varchar(128)	**DEPRECATED.**		
saved_direct	varchar(128)	This field specifies if the datum/metric is retrieved directly or if it is saved in some datastore.		
kb_based	varchar(128)	This field specifies if the datum/metric is retrieved from the Km4City Knowledge Base or not.		
sm_based	varchar(128)	This field specifies if the datum/metric is based on the DISIT Service Map (for instance, if it is retrieved by calling the Service Map Smart City API).		
user	varchar(128)	**DEPRECATED.**		
widgets	varchar(128)	**DEPRECATED.**		
parameters	varchar(512)	General parameters (e.g.: URI for API calls) and information useful to represent the specific datum/metric.		
healthiness	varchar(128)	This field specifies if the specific datum/metric is “healthy”, according to a specific scheduled process running in background which check the “healthiness” of each data.		
microAppExtServIcon	varchar(100)	Icon name for External Services and Apps.		
lastCheck	datetime	Date of last healthiness check for the specific datum/metric.		
ownership	varchar(64)	This field specifies if the datum/metric is public or private.		

Data

Each row of this table represents the actual values measured/produced by all the metrics.

Data				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
IdMetric_data	varchar(45)	Auto-incremental id of the current metric.	PRIMARY	
computationDate	datetime	Datetime of the computation for the current metric.	PRIMARY	
value_num	double	If the value type is numeric (not percentage), then the value is stored in this field.		
value_perc1	double	If the value type is percentage, then the value is stored in this field.		
value_perc2	double	In case a metric compute or relies on a second percentage value for the same sample data, then it is stored in this field. **NOT USED**.		
value_perc3	double	In case a metric compute or relies on a third percentage value for the same sample data, then it is stored in this field. **NOT USED**.		
value_text	varchar(300)	If the value type is string, then the value is stored in this field.		
quant_perc1	int(11)	**OLD – NOT USED ANYMORE.** In case the value type is a percentage, the dividend was stored in this field.		
quant_perc2	int(11)	**OLD – NOT USED ANYMORE.** In case a metric compute or relies on a second percentage, the second dividend was stored in this field.		
quant_perc3	int(11)	**OLD – NOT USED ANYMORE.** In case a metric compute or relies on a third percentage, the third dividend was stored in this field.		
tot_perc1	int(11)	**OLD – NOT USED ANYMORE.**		
tot_perc2	int(11)	**OLD – NOT USED ANYMORE.**		
tot_perc3	int(11)	**OLD – NOT USED ANYMORE.**		
series	text	If the value type is Series (each sample data is actually a serie – matrix), the value is stored in this field.		

DataSource

Each row of this table describe the different data sources used for ingesting data in the platform.

DataSource				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
intId	int(9)	Auto-incremental integer ID.	PRIMARY	
Id	varchar(45)	Unique name as Identifier for the current data source.	Id_UNIQUE	
url	varchar(200)	URL of the current data source.		
database	varchar(100)	Name of the database		
username	varchar(100)	Username to access the current data source.		
password	varchar(100)	Password to access the current data source.		
databaseType	varchar(100)	Type of data source (MySQL, RDF etc…).		

Descriptions

Each row of this table describes the different metrics igested by the platform.

Descriptions				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
intId	int(9)	Auto-incremental integer ID.	PRIMARY	
IdMetric	varchar(45)	Unique name as Identifier for the current metric.	IdMetric_UNIQUE	
description	longtext	A description of the current metric.		
status		Active / Not Active.		
query	longtext	Query used to retrieve the current metric.		
query2	longtext	**Not used.**		
queryType	varchar(300)	SQL, SPARQL, …		
metricType	varchar(100)	Integer, float, …		
frequency	varchar(100)	Frequency (measured in milliseconds) at which the current metric is retrieved.		
processType	varchar(45)	e.g.: {API, JVNum1, checkStatus, responseTime, …}		
area	varchar(300)	Classification of the current metric.		
source	varchar(300)	e.g.: {Disit, Servicemap, Wallet, COMUNE DI FIRENZE, Osservatorio Regionale Mobilità e Trasporti, …}		
description_short	longtext	A short description of the current metric.		

dataSource	varchar(100)	Km4CityRDF, DataCenterMachine166, Km4CityApp, DisitDataCenter2, Km4CityRTData, WifiStreaming, TwitterTV, Km4CityEngager-UserProfile,		
storingData	int(11)	0/1		
municipalityOption	int(11)	0/1		
timeRangeOption	int(11)	0/1		
field1Desc	varchar(45)	**Not used.**		
field2Desc	varchar(45)	**Not used.**		
field3Desc	varchar(45)	**Not used.**		
oldData	int(1)	**DEPRECATED.**		
sameDataAlarmCount	int(11)	Number of same data to be received in order to fire and send an alarm message.		
oldDataEvalTime	int(11)	**DEPRECATED.**		
hasNegativeValues	int(1)	0/1		
process	varchar(45)	DashboardProcess / HttpProcess		
threshold	double	**Not used.**		
thresholdEval	varchar(45)	**Not used.**		
boundToMetric	varchar(45)	**Not used.**		
status_HTTPRetr	varchar(45)	???		
username_HTTPRetr	varchar(45)	**Not used.**		
password_HTTPRetr	varchar(45)	**Not used.**		

IdDashDailyAccess

Each row of this table ...

IotApplications				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
IdDashboard	int(11)	Unique Identifier ID of the current dashboard.	UniqueKey	Config_dashboard.Id / YES Config_widget_dashboard.Id / NO DashboardsViewPermissions.IdDashboard / NO
name_dashboard	varchar(300)			
date	date		UniqueKey	
nAccessPerDay				
nMinutesPerDay				

JobAreas

Each row of this table ...

JobAreas				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental integer ID.	PRIMARY	
schedulerId	Int(11)	???	schedulerId	
name	varchar(45)	e.g.: “Stato linee ATAF”, “Check RT”, “Eventi a Firenze”, “Parcheggi”, “Previsioni Meteo” ecc…		

MainMenu

Each row of this table ...

MainMenu				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental integer ID.	PRIMARY	
linkUrl	varchar(200)	Absolute/relative path for main menu, “submenu” for submenus.		
linkId	varchar(200)			
icon	varchar(200)			
privileges	text			
userType	varchar(45)			
externalApp	varchar(3)			
openMode	varchar(45)	e.g.: “submenu”, “samePage”, “iframe”, “newTab”		
iconColor	varchar(45)			
pageTitle	varchar(200)			
domain	varchar(2)			
menuOrder	varchar(2)			

MainMenuSubmenus

Each row of this table ...

MainMenuSubmenus				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental integer ID.	PRIMARY	
menu	int(11)			
linkUrl	varchar(200)			
linkId	varchar(200)			
icon	varchar(200)			
text	varchar(200)			
privileges	text			
userType	varchar(45)			
externalApp	varchar(3)			
openMode	varchar(45)			
iconColor	varchar(45)			
pageTitle	varchar(200)			
menuOrder	varchar(2)			

MobMainMenu

Each row of this table ...

MobMainMenu				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental integer ID.	PRIMARY	
linkUrl	varchar(200)	Absolute/relative path for main menu, “submenu” for submenus.		
linkId	varchar(200)			
icon	varchar(200)			
privileges	text			
userType	varchar(45)			
externalApp	varchar(3)			
openMode	varchar(45)	e.g.: “submenu”, “samePage”, “iframe”, “newTab”		

iconColor	varchar(45)			
pageTitle	varchar(200)			
domain	varchar(2)			
menuOrder	varchar(2)			

MobMainMenuSubmenus

Each row of this table ...

MobMainMenuSubmenus				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental integer ID.	PRIMARY	
menu	int(11)			
linkUrl	varchar(200)			
linkId	varchar(200)			
icon	varchar(200)			
text	varchar(200)			
privileges	text			
userType	varchar(45)			
externalApp	varchar(3)			
openMode	varchar(45)			
iconColor	varchar(45)			
pageTitle	varchar(200)			
menuOrder	varchar(2)			

NodeRedMetrics

Each row of this table represents a metric created/managed by Node-RED applications.

NodeRedMetrics				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***

id	int(11)	Auto-incremental integer ID.	PRIMARY	
name	varchar(400)		PRIMARY2	
metricType	varchar(400)		PRIMARY2	
user	varchar(400)		PRIMARY2	
shortDesc	varchar(400)			
fullDesc	varchar(400)			
appId	varchar(400)	Unique ID of the Node-RED application, if the current widget is associated with a Node-RED app.		NodeRedInputs.appId / NO Config_widget_dashboard,appId / NO
flowId	varchar(400)	Unique ID of the Node-RED flow, if the current widget is associated with a Node-RED app (a Node-RED app can contain multiple Node-RED flows).		NodeRedInputs.flowId / NO Config_widget_dashboard.flowId / NO
flowName	varchar(400)	Name of the Node-RED flow represented by *flowId.*		
nodeId	varchar(400)	Unique ID which is automatically assigned by Node-RED to each Node-RED block.		NodeRedInputs.nodeId / NO Config_widget_dashboard.nodeId / NO
httpRoot	varchar(400)	Realtive path of the Node-RED app, e.g.: "/nodered/nr25"		

NodeRedInputs

Each row of this table ...

NodeRedInputs				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental integer ID.	PRIMARY	
name	varchar(100)		name_UNIQUE	
valueType	varchar(45)			
user	varchar(100)			
startValue	text	e.g.: "Off", 0		
domainType	varchar(100)	e.g.: "geolocator", "impulse", "singleNumericValue"		
minValue	double			
maxValue	double			
offValue	varchar(200)	e.g.: "off", null		
onValue	varchar(200)	e.g.: "on", "Red", "Green", "Yellow", …,		
dataPrecision	int(11)	**Not used.**		
endPointPort	int(8)	e.g.: "1895", "1880"		
endPointHost	varchar(45)			
httpRoot	varchar(45)	Realtive path of the Node-RED app, e.g.: "/nodered/nr25"		

appId	varchar(400)	Unique ID of the Node-RED application, if the current widget is associated with a Node-RED app.		NodeRedMetrics.appId / NO Config_widget_dashboard,appId / NO
flowId	varchar(400)	Unique ID of the Node-RED flow, if the current widget is associated with a Node-RED app (a Node-RED app can contain multiple Node-RED flows).		NodeRedMetrics.flowId / NO Config_widget_dashboard.flowId / NO
flowName	varchar(400)	Name of the Node-RED flow represented by *flowId.*		
nodeId	varchar(400)	Unique ID which is automatically assigned by Node-RED to each Node-RED block.		NodeRedMetrics.nodeId / NO Config_widget_dashboard.nodeId / NO

OperatorEvents

Each row of this table represents the status of a single event from operators such as FirstAid...

OperatorEvents				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(11)	Auto-incremental integer ID.	PRIMARY	
time	varchar(45)			
personNumber	int(11)			
lat	double			
lng	double			
codeColor	varchar(45)			
user	varchar(100)			

Widgets

Each row of this table represents the different widget types (and NOT their single instances) with some default parameters and information.

Widgets				
Field Name	***Format***	***Description***	***Key/Index***	***Ext. References / Foreign Keys***
id	int(9)	Auto-incremental integer ID.	PRIMARY	
id_type_widget	varchar(150)	Unique name identifier of the current widget type.	id_type_widget_UNIQUE	
source_php_widget	varchar(150)	PHP Source file name of the current widget type.		

min_row	int(11)	Minimum number of rows for displaying the current widget type on a dashboard.		
max_row	int(11)	Maximum number of rows for displaying the current widget type on a dashboard.		
min_col	int(11)	Minimum number of columns for displaying the current widget type on a dashboard.		
max_col	int(11)	Maximum number of columns for displaying the current widget type on a dashboard.		
widgetType	varchar(150)	Type of the metric read/written by the current widget type.		
unique_metric	varchar(150)			
numeric_rangeOption	int(11)			
number_metrics_widget	int(11)	Number of metrics used (usually in reading mode) by the current widget type.		
color_widgetOption	int(11)			
dimMap	text			
widgetCategory	varchar(45)	e.g.: “dataviewer” (widget displaying – READ – data collected by the platform), “actuator” (widget acting – WRITE – on city sensors, IOT etc.).		
isNodeRedSender	varchar(3)	‘Yes’ for actuators, ‘No’ for data viewer ???		
domainType	text			
defaultParameters	text			
hasTimer	varchar(3)			
hasChartColor	varchar(3)			
hasDataLabels	varchar(3)			
hasChartLabels	varchar(3)			
hasTimeRange	varchar(3)			
hasCartesianPlane	varchar(3)			
hasChangeMetric	varchar(3)	‘Yes’ if it is possible, for the current widget, to change its metric directly from context menu; ‘No’ otherwise.		
hasAddMode	varchar(3)	Valid only for widgetMap: it specifies if the widgetMap can be in additive mode (that is, showing more resources for multiple selectors at once) or not.		

WidgetsIconsMap

Each row of this table represents the icon used in the Dashboard Wizard to represent each widget type, and other parameters and information useful to instantiate them from the Dashboard Wizard.

WidgetsIconsMap

Field Name	Format	Description	Key/Index	Ext. References / Foreign Keys
id	int(9)	Auto-incremental integer ID.	PRIMARY	
mainWidget	varchar(150)			
targetWidget	varchar(150)	When present, this is the widget piloted by the main widget (e.g.: an external content, for example a map, which can be piloted by a selector, representing different geolocated resources, or an iFrame which can be piloted by a Selector Web, showing different web pages…)		
snap4CityType	varchar(400)			
icon	varchar(150)			
mono_multi	varchar(150)			
description	Text			
available	varchar(10)			
sm_based	varchar(3)			
defaultParametersMainWidget	text			
defaultParametersTargetWidget	text			
hasMainWidgetFactory	varchar(3)	{“yes”, “no”}		
hasTargetWidgetFactory	text	{“yes”, “no”}		
comboName	varchar(100)			
widgetCategory	varchar(45)			

Dashboard DB Schema