



User manual

Screen Composer & SignCom

1.13.10 001A



Legal information

Screen Composer & SignCom 1.13.10 (001A_en)

© 2021 INNES

Rights and Responsibilities

All rights reserved. No part of this manual may be reproduced in any form or by any means without written permission of the publisher. Products and services mentioned herein may be trademarks and/or registered trademarks of their respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and author assume no responsibility for errors or omissions, or for damages resulting from the use of this document, or from the use of the programs and source code that may accompany it. In no event shall the publisher and author be liable for any loss of profit or other commercial loss caused or contributed to by the use of this document.

Product information

The design and specifications of the product are subject to change at any time and INNES reserves the right to change them without notice. This includes the hardware, embedded software and this manual, which should be considered as a general guide to the product. The accessories supplied with the product may differ slightly from those described in this manual, depending on the developments of the various suppliers.

Table of contents

Part I : Introduction

Introduction	1.1
--------------	-----

Part II : Screen Composer installation and settings

Installation	2.1
Gekkota devices configuration to work with Screen Composer G4	2.1.1
Qeedji System AOSP devices configuration to work with Screen Composer G4	2.1.2
Getting started	2.2
Screen Composer > About	2.2.1
Screen Composer > Preferences > Repositories	2.2.2
Screen Composer > Preferences > Network	2.2.3
Screen Composer > Preferences > Middlewares	2.2.4
Screen Composer > Preferences > Licenses and Apps	2.2.5
App installation	2.2.5.1
App license keys installation	2.2.5.2
Screen Composer > Preferences > Advanced	2.2.6
File > Project settings > Devices	2.2.7
File > Project settings > Project and App	2.2.8
File > Project settings > Publication	2.2.9
File > Project settings > Tasks	2.2.10

Part III : Project

SignCom new project	3.1
Medias	3.2
Medias > Preview and metadata	3.2.1
Medias > URI	3.2.2
Medias > Playfolder	3.2.3
Medias > Playlist	3.2.4
Medias > Slide	3.2.5
Medias > Create from a model	3.2.6
Medias > Text file	3.2.7
Medias > Calendar	3.2.8
Fonts	3.3
Variables	3.4
Custom variable	3.4.1
Device information variable	3.4.2
Date and time variable	3.4.3
Services accounts	3.5
Devices	3.6
SignCom > Visual channel	3.7
SignCom > Manifest	3.8
SignCom > Properties	3.9

Part IV : Publication

App publication	4.1
Middleware publication	4.2

Part V : Contacts

Contacts	5.1
----------	-----

Part VI : Appendix

Appendix: Content model short description	6.1
Appendix: Medias ending management for HTML widgets and URI based on models	6.2
Appendix: supported inputs and buttons properties to support identifier and password self-filling and auto-confirmation in a Web page form	6.3
Appendix: Azure AD portal	6.4
Appendix: Azure AD Application PowerShell module	6.5
Appendix: Teams	6.6
Appendix: discovery & publication timings	6.7

Part I

Introduction

1.1 Introduction

The SignCom App is a powerful solution allowing to realize thanks to its calendar a channel programmation 24/7/365 by dragging easily many types of media, displayed in the different regions of the screen. Transition between scenes and medias are supported allowing to realize your own channel. The SignCom App allows also to preview the content of the channel programmation before publishing the App on the devices. The SignCom App can be configured and published on a network of screens with INNES Screen Composer G4 . For further information, refer to the chapter § [Screen Composer > Installation](#).

Part II

**Screen Composer installation and
settings**

2.1 Installation

INNES Screen Composer G4 Desktop is a single-user and single-computer station application. In this user manual, each INNES Screen Composer G4 project, whose extension is `.isz`, is based on a SignCom App product.

Once your SignCom App product is configured, INNES Screen Composer G4 allows to publish it on several devices.

After a first installation, INNES Screen Composer G4 embeds no App product.

Some core INNES Screen Composer G4 license keys must be installed in order to have the publishing tokens required to publish to devices.

As soon as an SignCom App is published on a device, a core license key is automatically associated with that device, using its MAC id.

If you use a pack of 20 core license keys or a pack of 5 core license keys, the total number of core license keys must be greater than or equal to the number of devices registered in your Screen Composer, regardless of the number of Apps used.

The using of the SignCom App requires also the installation of an SignCom App license key.

Prerequisite

INNES Screen Composer G4 must be installed on a recent computer:

- OS: MS-Windows 10,
- Processor: Core i5/i7,
- DDR: 4 GB (or more),
- HDD: 160 GB (or more).

The firewall of this computer must allow the port 80 (http) so that INNES Screen Composer G4 can publish on the devices, available by default on the port 80.

Supported OS/devices

INNES Screen Composer G4 can work with :

- Qeedji System AOSP 9.10.14 (or above) for:
 - AMP300 (media player),
 - TAB10s (tablet),
- Gekkota OS 4.14.14¹ (or above) for:
 - DMB400 (media player),
 - SMA300 (media player),
- Gekkota OS 5.11.10 (or above) for :
 - DMB300 (media player),
 - DMB400 (media player),
- Gekkota RT 4.14.10¹ (or above) for:
 - MS-Windows computer,
- Gekkota OS 3.12.57 (or above) for:
 - DMB300 (media player),
 - DMC200 (media player),
- Gekkota RT 3.12.59 (or above) for:
 - MS-Windows computer.

¹ Gekkota 4 OS/RT version under 4.13.XX (or below) may be not compatible if the SignCom App contains a playfolder with Microsoft 365 services account .

Installation

To install INNES Screen Composer G4 , execute `screen-composer-nt_ia32-setup-4.<yy>.<zz>.exe` .

Select the appropriate user interface language among the suggested INNES Screen Composer G4 languages:

- English,
- French,
- German.

160 MB disk memory are first required to be able to install INNES Screen Composer G4. Then some additional disk space will depend on the space required for your project repository.

The default directory for the installation is:

- `C:\Program Files (x86)\INNES Screen Composer G4` .

The default shortcuts located in the Windows startup menu is:

- INNES Screen Composer G4 .

Launch

When the installation is completed, launch INNES Screen Composer G4 by using the:

- MS-Windows startup menu,
- MS-Windows cortana.



INNES Screen Composer G4

Language

To change the language of the INNES Screen Composer G4 user interface, uninstall INNES Screen Composer G4 and install it again by choosing among the supported languages described before.

Uninstallation

Use the Add or remove programs tool in MS-Windows System settings to remove the INNES Screen Composer G4 program from your computer.

2.1.1 Gekkota devices configuration to work with Screen Composer G4

Gekkota 4 devices

The Gekkota 4 devices are the devices that have a Gekkota OS or Gekkota RT middleware whose version is 4.<yy>.<zz>. To work with INNES Screen Composer G4, ensure that your devices have a Gekkota OS 4.13.10 firmware version (or above).

To work as devices with INNES Screen Composer G4, ensure that your dedicated MS-Windows computers have Gekkota RT 4.13.15 version (or above) and fill the following configuration:

- OS: MS-Windows 10,
- Processor: Core i5/i7,
- DDR: 4 GB (or more),
- HDD: 160 GB (or more).

To install Gekkota RT on a dedicated MS-Windows computer, download the Gekkota RT middleware from the [Innes Web site](#), execute the `gekkota_rt-nt_ia32-setup-4.<yy>.<zz>.exe` installer on the dedicated MS-Windows computer. After the installation is completed, the Gekkota RT may start up automatically and display an information message *Information - No content*.

■ *Depending on the computer, the Gekkota start-up duration may be few minutes.*

Provide a suitable power supply to the device then wait the LED status is blinking once every four seconds. If coming straight from factory, the `test` card content should be displayed on the screen.

A Gekkota G4 device can be configured with its Web user interface by using this URL `http://<gekkota_4_device_ip_addr>/` with a recent version of a Web browser like Google Chrome, Mozilla Firefox and Microsoft Edge (Chromium).

■ *The default login credentials to connect to the Web user interface are admin / admin.*

Then click on the `Administration Console` button to display the device configuration Web user interface.

■ *After having made some parameters modification, the device must be restarted so that the modifications are fully reflected.*

For further information, refer to the appropriate user manual.

Gekkota OS G4 devices user manuals
DMB400 user manual
SMA300 user manual
DME204 user manual

Gekkota 5 devices

The Gekkota 5 devices are the devices that have a Gekkota OS middleware whose version is 5.<yy>.<zz>. To work with INNES Screen Composer G4, ensure that your devices have a Gekkota OS 5.10.11 firmware version (or above).

A Gekkota 5 device can be configured with its web-based configuration interface using this URL `http://<gekkota_5_device_ip_addr>/` with a recent version of a web browser like Google Chrome, Mozilla Firefox and Microsoft Edge.

■ *The default login credentials to connect to the device configuration Web user interface are admin / admin.*

■ *After having made some parameters modification, the device must be restarted so that the modifications are fully reflected.*

Gekkota 3 devices

The Gekkota 3 devices are the devices that have a Gekkota OS or Gekkota RT middleware whose version is 3.<yy>.<zz>. To work with INNES Screen Composer G4, ensure that your devices have a Gekkota OS 3.12.57 firmware version (or above).

To work as devices with INNES Screen Composer G4, ensure that your dedicated MS-Windows computers have Gekkota RT 3.12.59 version (or above) and fill the following configuration:

- OS: MS-Windows 10,
- Processor: Core i5/i7,
- DDR: 4 GB (or more),
- HDD: 160 GB (or more).

To install Gekkota RT on a dedicated MS-Windows computer, download the Gekkota RT middleware from the [Innes Web site](#), execute the `gekkota_rt-nt_ia32-setup-3.<yy>.<zz>.exe` installer on the dedicated MS-Windows computer. After the installation is completed, the Gekkota RT may start up automatically and display the content "Information - No content".

■ *Depending on the computer, the Gekkota start-up duration may be few minutes.*

A Gekkota 3 device can be configured with its Web user interface. Given that Flash technology is not supported in Web browsers anymore, the Web user interface of this device generation can only be opened using a third part supporting this technology (for example [Gekkota Desktop](#)).

■ *The default login credentials to connect to the Web user interface are admin / admin.*

■ *After having made some parameters modification, the device must be restarted so that the modifications are fully reflected.*

For further information, refer to the appropriate user manual.

Web UI: Configuration > Administrator

It is recommended that you enter a different `Hostname` value for each device. In case several tablets are located in different buildings or geographical locations, we recommend that you enter hostname values containing information about the building and the location (e.g. `Hall-RD-Paris-1`).

For security reasons, it may be useful to change the `Administrator name` and `password` login credentials to access to the device's WebDAV server.

☞ If modified, store previously the login credentials because they are required to register properly the devices in INNES Screen Composer G4.

Web UI: Configuration > WLAN

If the device supports the `WIFI` option and needs to access to the `WIFI` network, activate and configure properly the WLAN interface:

- `DHCP`,
- `IP address`,
- `subnet mask`,
- `gateway`.

☞ Activating temporarily the `Test Card` allows to check whether this network interface STATE is up. It is also possible to view this status in the `Network` panel of the `Information` menu.

Web UI: Configuration > LAN

If the device needs to access to the `LAN` network, activate and configure properly the LAN interface:

- `DHCP`,
- `IP address`,
- `subnet mask`,
- `gateway`.

☞ Activating temporarily the `Test Card` allows to check whether this network interface STATE is up. It is also possible to view this status in the `Network` panel of the `Information` menu.

Web UI: Configuration > Output

When your SignCom App is configured in a portrait resolution, it is required to apply a 90° (or 270°) rotation to the media player output as well.

☞ The 90°, 180° and 270° rotation is supported on media-players but is not supported on the tablets.

Web UI: Configuration > App

Is it required that the device `App` mode is configured in `Push WebDAV`.

Web UI: Configuration > License

The `license` key is already registered in the device when it is coming straight from factory.

☞ When using Gekkota5 middleware, the license key value specific to the Gekkota 5 middleware is not provided. For further information, contact `sales@qeeji.tech`.

☞ The device license key has been provided by email when the device has been ordered then delivered. In case the license key has been erased and lost, another device license key can be obtained by sending an email to `License@qeedji.tech` by specifying in the subject of the message the MAC address value and the serial number value of the device.

Web UI: Configuration > Servers

It is advised that the device is on time. The `date` and `time` is set automatically thanks to a Web NTP server .

If ever the device could not be allowed to access to a Web NTP server , a `NTP server` needs to be implemented on your local network to provide the `date` and `time` to the device.

Web UI: Configuration > Date and time

It is advised that the device is on time. Set an appropriate value for `Timezone`, `Date` and `Time`.

☞ When the date and time is modified, the device restarts automatically once so that the time modification is applied as soon as possible.

Web UI: Maintenance > Test Card

When the device configuration is completed, the `Test Card` must be deactivated to see the SignCom App running on the screen.

☞ Activating temporarily the `Test Card` allows to check whether the STATE of a network interface is up.

Web UI: Maintenance > Tools

It could be required to install some appropriate *trusted certificate* when the SignCom App must play some medias hosted on remote servers available with the `https` scheme and requiring certificate.

2.1.2 Qeedji System AOSP devices configuration to work with Screen Composer G4

The Qeedji System AOSP devices are the devices that have a AQS operating system whose version is 9.<yy>.<zz>. To work with INNES Screen Composer G4 , ensure that your devices have a AQS 9.10.15 firmware version (or above).

In case using a AMP300 device, provide a suitable power supply to it, through the LAN/PoE RJ45 connector, the POWER IN USB-C or the USB 1 USB-C connector if your display device supports USB-C supply, then wait the LED status is blinking once every four seconds. If coming straight from factory, the device should display the test card content. In case using a TAB10s device, provide a suitable power supply to it then wait the device is displaying the content of the AQS desktop on the screen.

An Qeedji System AOSP device can be configured with its Web user interface by using this URL [http://<aqs_device_ip_addr>/](http://<aqs_device_ip_addr>) with a recent version of a Web browser like Google Chrome , Mozilla Firefox and Microsoft Edge (Chromium) .

- The default login credentials for the Administration User connection profile, allowing to connect to the device configuration Web user interface, are admin / admin .

Then click on the Administration button to display the device configuration Web user interface.

- After having made some parameters modification, the device must be restarted so that the modifications are fully reflected.

For further information about the connection to the device configuration Web user interface, refer to the appropriate user manual.

AQS device user manuals
AMP300 user manual
TAB10s user manual

Web UI: Configuration > Administrator

The default hostname is TAB10s for TAB10s device and AMP300 for AMP300 device. It is recommended that you enter a different hostname value for each device. In case several devices are located in different buildings or geographical locations, we recommend that you enter hostname values containing information about the building and the location (e.g. Hall-RD-Paris-1).

- For security reasons, it is advised to change the login credentials of the Administration User connection profile to access to the device configuration Web user interface.
- If modified, store preciously the login credentials because they are required in INNES Screen Composer G4 to communicate with the devices.

Web UI: Configuration > LAN1

Provide the network connectivity to the device by plugging an Ethernet cable then ensure that the device has a consistent IP address value.

- If the tablet needs to access to the LAN network, install the appropriate PoE to USB-C adapter allowing to provide IP network connectivity to the tablet then ensure that the tablet has a consistent IP address.
- Activating temporarily the Test Card App launching at start-up allows to check whether the LAN1 network interface (I/F) is ON and whether the network interface STATE is up. It is also possible to view this status in the Network panel of the Information menu.

Web UI: Configuration > WLAN1

If the device needs to access to the WIFI network, activate and configure the WIFI on the device then ensure that the device has a consistent IP address value.

- Activating temporarily the Test Card App launching at start-up allows to check whether the WLAN1 network interface (I/F) is ON and whether the network interface STATE is up. It is also possible to view this status in the Network panel of the Information menu.

Web UI: Configuration > Output

In case using an AMP300 device, plug a display device on it through the USB 1 USB-C connector or through the HDMI connector. Set an output resolution supported by your display device.

- The 90°, 180° and 270° rotation is supported on AMP300 devices but is not supported on the TAB10s devices.

Web UI: Configuration > App

After having published the SignCom App on the device, the device installs the SignCom App then restarts automatically once. After this device restart, the couple of App name (SignCom, SignCom UI) must appear in the table of activable Apps.

- It is advised to inactivate all other unuseful Apps.

Web UI: Configuration > Server

Enter the appropriate DNS server and gateway values to access to the Web contents.

The date and time is set automatically thanks to a Web NTP server .

- If ever the device could not be not allowed to access to a Web NTP server , a NTP server needs to be implemented on your local network to provide the current date and time to the device.

Web UI: Configuration > Certificates

It could be required to install some appropriate *trusted certificate* when the SignCom App must play some medias hosted on remote servers available with the `https` scheme and requiring certificate.

Web UI: Configuration > Date & Time

It is advised that the device is on time. The `date` and `time` is set automatically thanks to a Web NTP server . It is advised that the timezone is properly set.

Web UI: Maintenance > Test Card

When the device configuration is completed, the `Test Card` App launching at start-up must be deactivated to see the SignCom App running on the screen.

Web UI: Maintenance > Firmware

To work with INNES Screen Composer G4 , the firmware version of Qeedji System AOSP for TAB10s device or AMP300 device, must be 9.10.15 (or above).

Kiosk mode

When delivered straight from factory, the device is in `native` mode. The SignCom App is executed automatically in `kiosk` mode, making that the `back`, `menu` and `swipe` AOSP features are inactivated and so can not be used. You can exit temporally from the `kiosk` mode by pressing on the `system` button of the AQS device.

2.2 Getting started

To work with INNES Screen Composer G4 , launch INNES Screen Composer G4 then:

- [install once your INNES Screen Composer G4 core license key with the user email.](#)

To create a project in INNES Screen Composer G4 based on the SignCom App, you need to:

- [create a project](#),
- [define a default project repository](#) to store especially, in the Media tab of your project repository:
 - medias (images, videos, audios, MS-Powerpoint presentations, PDF documents, video URI),
 - Web TV URI ¹
 - Web pages,
 - playlists,
 - playfolders,
 - HTML widget (.maff, .wgt)²
 - Javascript script (.js),
- drop medias in the calendar at the appropriate date.

¹ Some Web TV URI can be created from URI models.

² Some HTML widget can be created from content models.

To publish your SignCom App on your devices, your need to add some devices to your project, choose the Publish the App to devices button value, and click on the button. For further information, refer to the chapter § File > Project settings > Devices.

To launch INNES Screen Composer G4 , search the INNES Screen Composer G4 Desktop application and launch it.



Welcome page

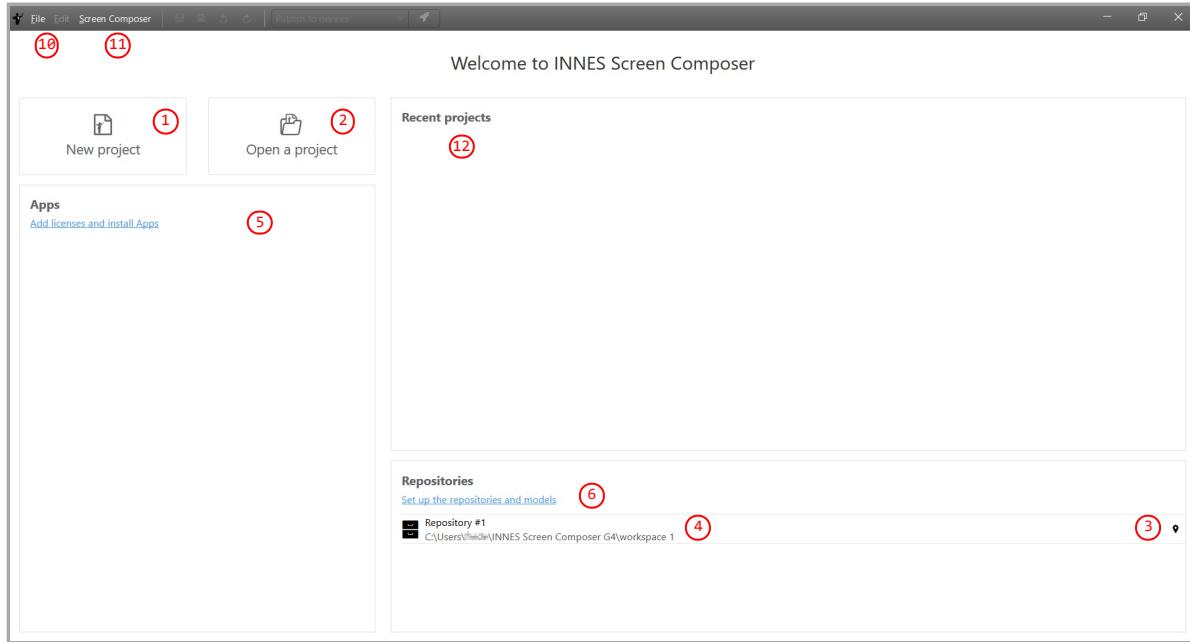
This is the welcome page when no `App` is installed, no `Project` is opened and no project repository is created.

The available buttons at this step are:

- New project **①**,
- Open project **②**,
- Reveal in the explorer **③** the default repository:
C:\Users\<username>\INNES Screen Composer G4\workspace 1\ **④**.

The available links at this step are:

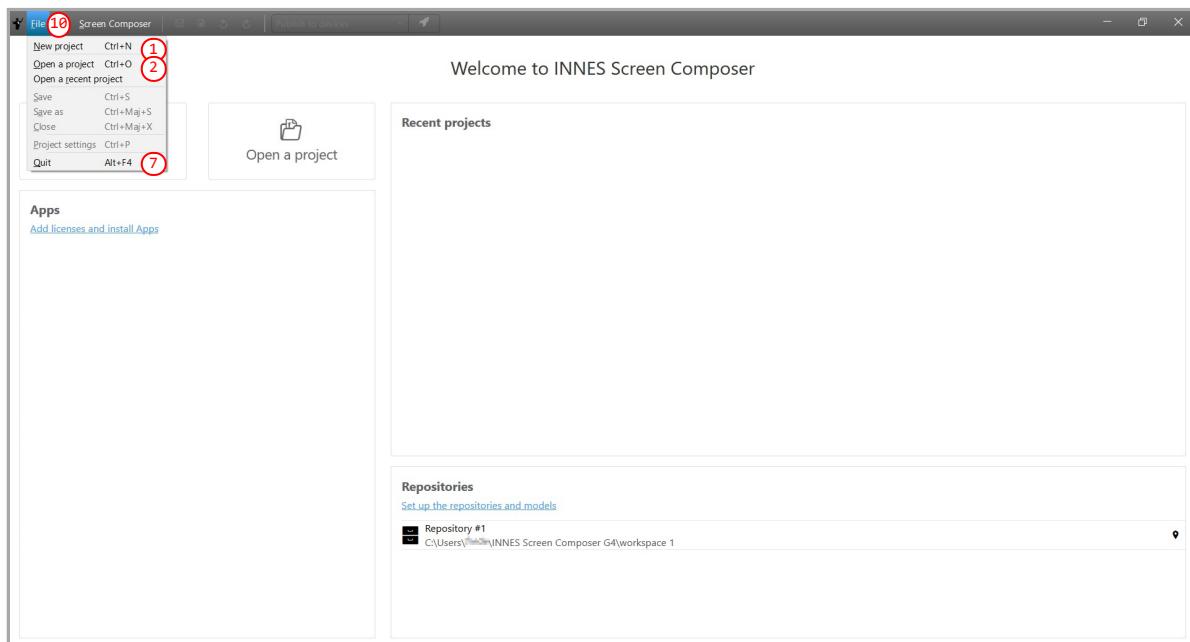
- Install licenses and install Apps **⑤**,
- Set up the repositories and models **⑥**.



The last opened projects will appear in the Recent projects **12** pane.

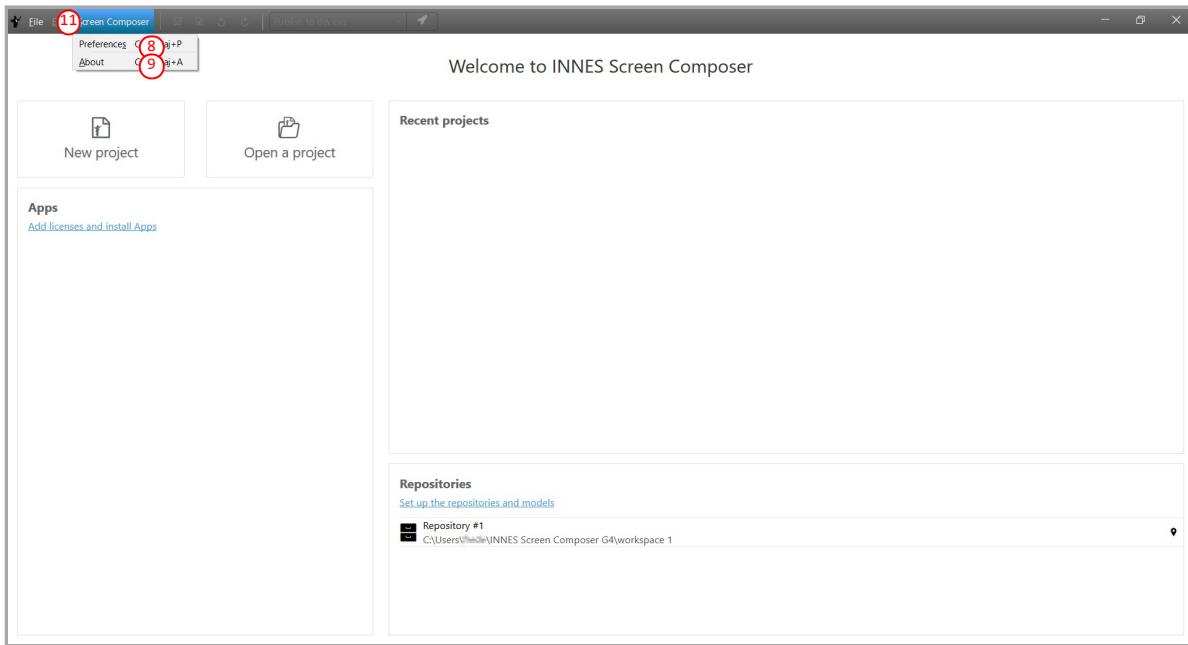
The available items, at this step, in the File **10** menu are:

- New project **①**,
- Open project **②**,
- Quit **⑦**.



The available items in the Screen Composer [\(11\)](#) menu are:

- Preferences [\(8\)](#)
- About [\(9\)](#).



About

The About pane allows to display information about INNES Screen Composer G4. For further information, refer to the chapter [§ Screen Composer > About](#).

Repositories

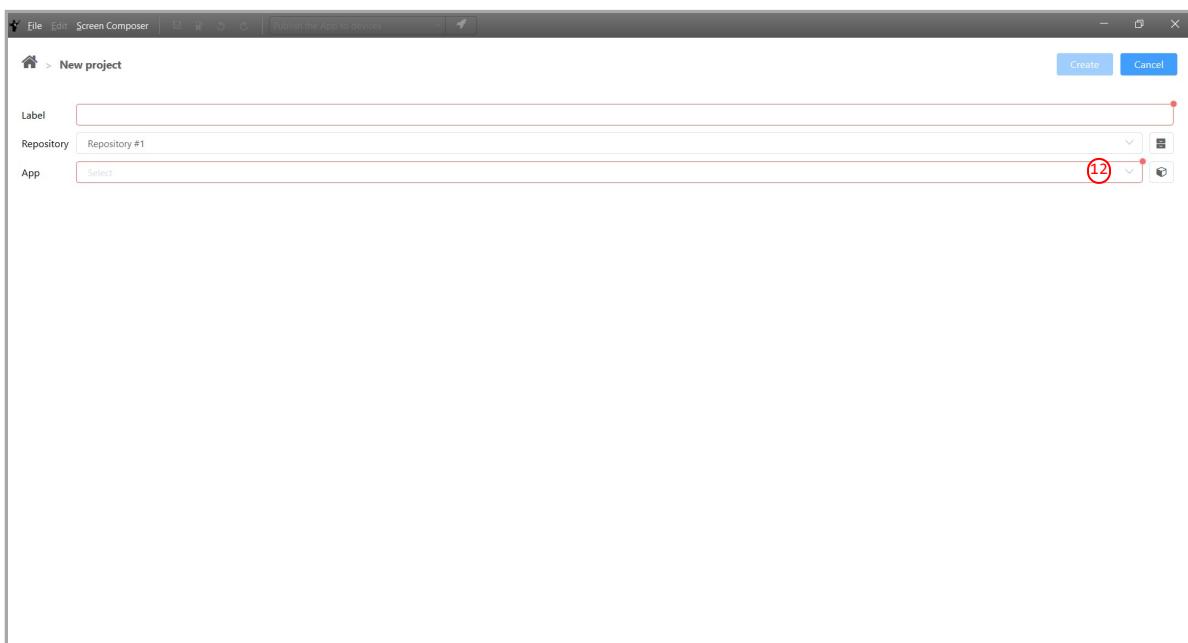
The default project repository for your project is:

C:\Users\<username>\INNES Screen Composer G4\workspace 1\

To create another project repository, refer to the chapter [§ Repositories](#).

Licenses and Apps

No SignCom App [\(12\)](#) is installed by default in INNES Screen Composer G4.



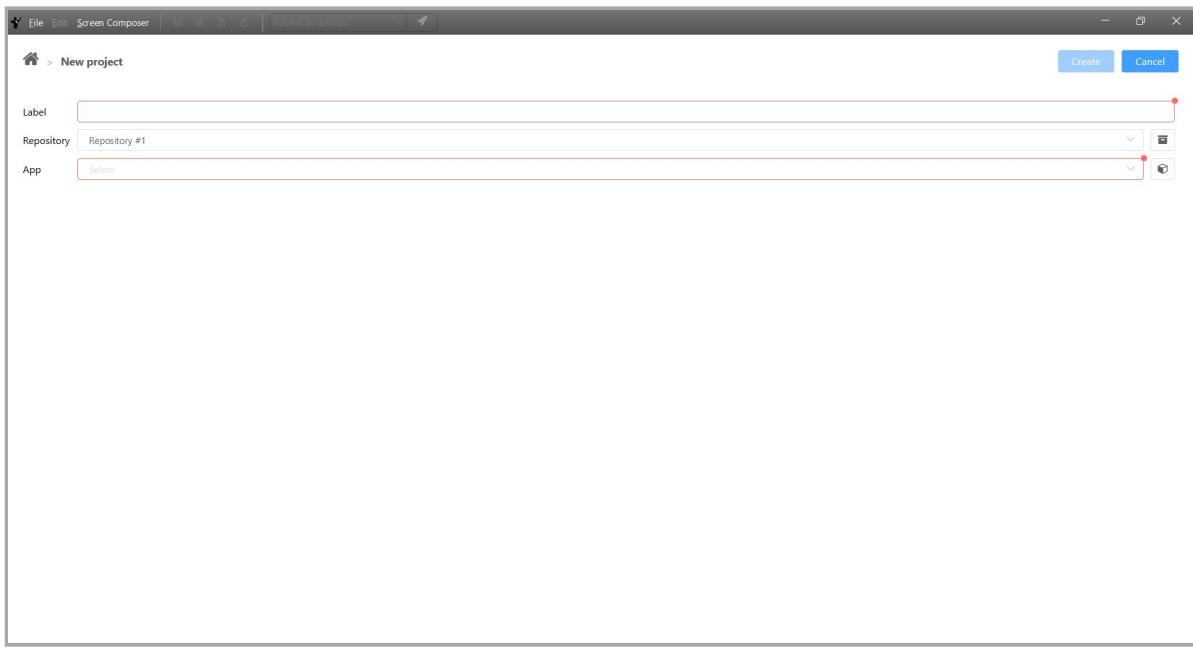
To install the SignCom App, refer to the chapter [§ SignCom App installation](#).

To install your SignCom App and your INNES Screen Composer G4 core license key with the user email, refer to the chapter [§ Screen Composer > Preferences > Licenses and Apps](#).

 The license keys used for Playzilla/Screen Composer G3 can be used also for SignCom/Screen Composer G4.

New project

To create a new INNES Screen Composer G4 project, click on the **New project** button (or **New project** item of the **File** menu).



Enter a label for your project and select the SignCom App on which the project is based.

- ☞ If there is no SignCom App available at this step, click on the **Edit Apps** button to install an App. For further information, refer to the chapter § [Screen Composer > Preferences > Licenses and Apps](#).
- ☞ Once the project is saved, it can be reopened with the **Open project** item of the **File** menu.

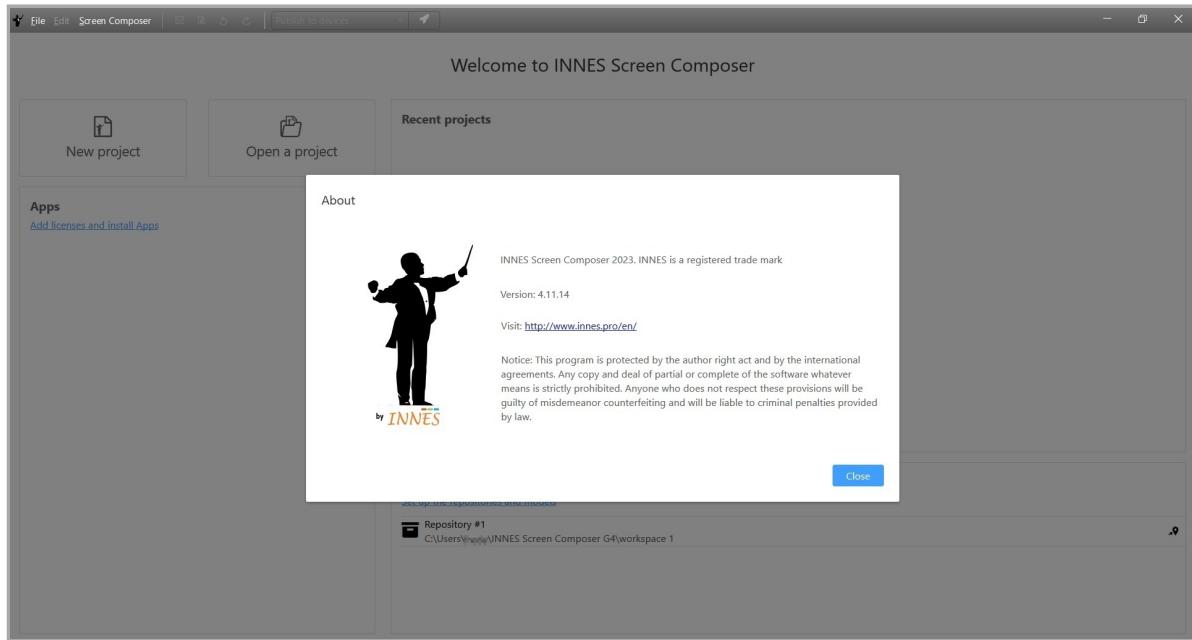
For further information, refer to the chapter § [SignCom new project](#).

2.2.1 Screen Composer > About

In the Screen Composer menu, click on the **About** item to see the version of INNES Screen Composer G4.

The **About** pane allows to display information about:

- INNES Screen Composer G4 version (e.g. 4.12.10),
- INNES Web site,
- Rights notice.



2.2.2 Screen Composer > Preferences > Repositories

A project repository must be mounted on a local directory of your MS-Windows file system allowing to store medias, content models and projects resources. It is required to define a default repository when creating your Screen Composer project.

INNES Screen Composer G4 can have different project repositories. But only one project repository is active per project at a time.

When created, a project repository contains this tree:

```
- /.db,  
- /.fonts,  
- /.medias,  
- /.models,  
- /.tmp,  
- .lock,  
- .uuids.
```

 It is recommended to use an empty directory for the repository. Indeed, in case you decide to remove this repository afterwards, the directory will be completely removed from your file system.

The /.medias directory is corresponding to  Media tab of the project repository.

The /.models directory is containing the content models and URI models.

 In this version, there is no specific view allowing to list all the content models and all the URI models.

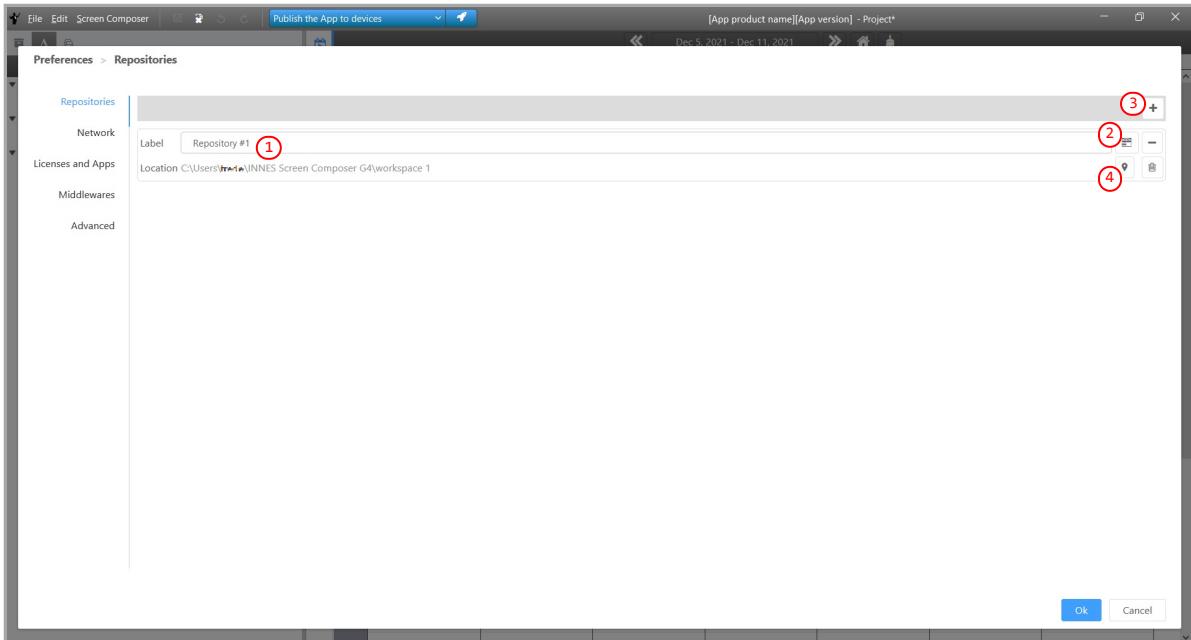
To show the available repositories, select the Preferences item of the Screen Composer menu then select the Repositories item.

Each project repository has a label  (1) which can be then modified afterwards. The default project repository is named Repository #1.

It is possible to create HTML widget from content model. To install the content models packs in your repository, download the appropriate content model packs (*Education, General, Traffic Info France, Signage*) (.zip) on your computer from the [INNES support Web site](#). Then click on the `Install models from your computer`  (2) button.

It is possible also to create URI from URI model. To install the URI models packs in your repository, download the appropriate URI model packs (*Device peripherals, Web TV channels, TV channels France*) (.zip) on your computer from the [INNES support Web site](#). Then click on the `Install models from your computer`  (2) button.

To add a new project repository, click on the  (3) button then select an appropriate directory.



It is possible also to switch to another project repository afterwards. For further information, refer to the chapter § [File > Project settings > Project and App](#).

 The repositories references are stored in the `prefs.js` file:

`C:\Users\<username>\AppData\Roaming\INNES Screen Composer G4\prefs.js`

 To open the directory location of your project repository in the MS-Explorer, click on the `Reveal in the explorer`  (4) button.

2.2.3 Screen Composer > Preferences > Network

In the Preferences item of the Screen Composer menu, select the Network item.

Set the toggle button to the right (blue color) to use a proxy server.

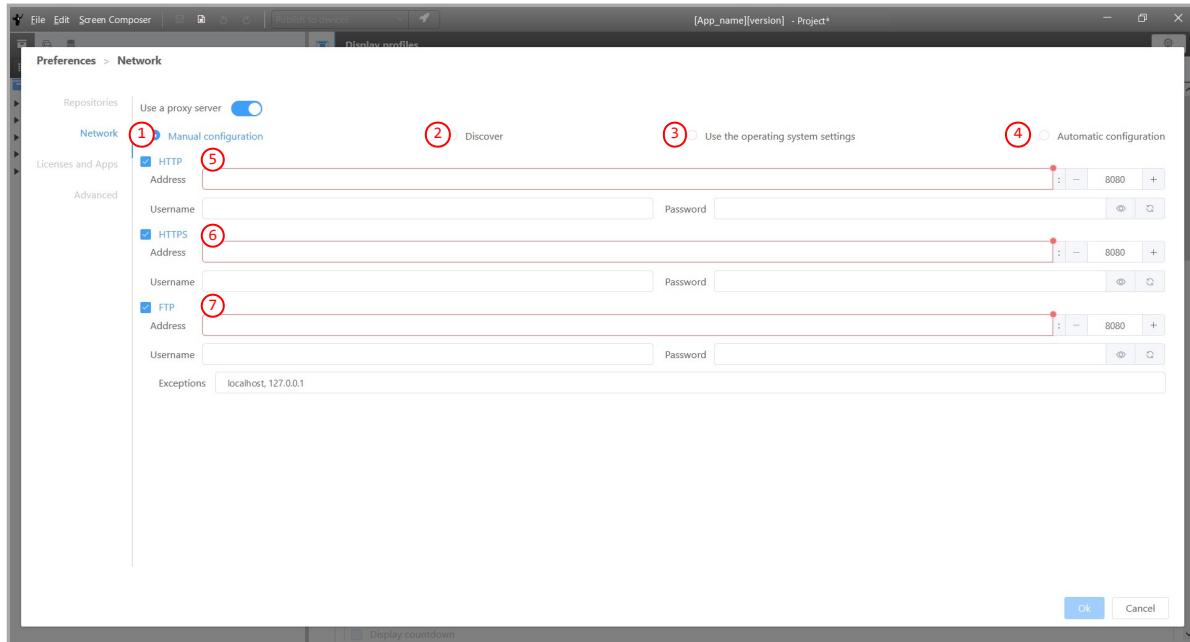
Several configurations are then possible:

- Manual configuration (1),
- Discover (2),
- Use the operating system settings (3),
- Automatic configuration (4).

Proxy > Manual configuration

Select the Manual configuration radio to enter the proxy configuration yourself:

- the proxy IP address and login credentials for HTTPS (5),
- the proxy IP address and login credentials for HTTP (6),
- the proxy IP address and login credentials for FTP (7),
- the domains exceptions .



Proxy > Discover configuration

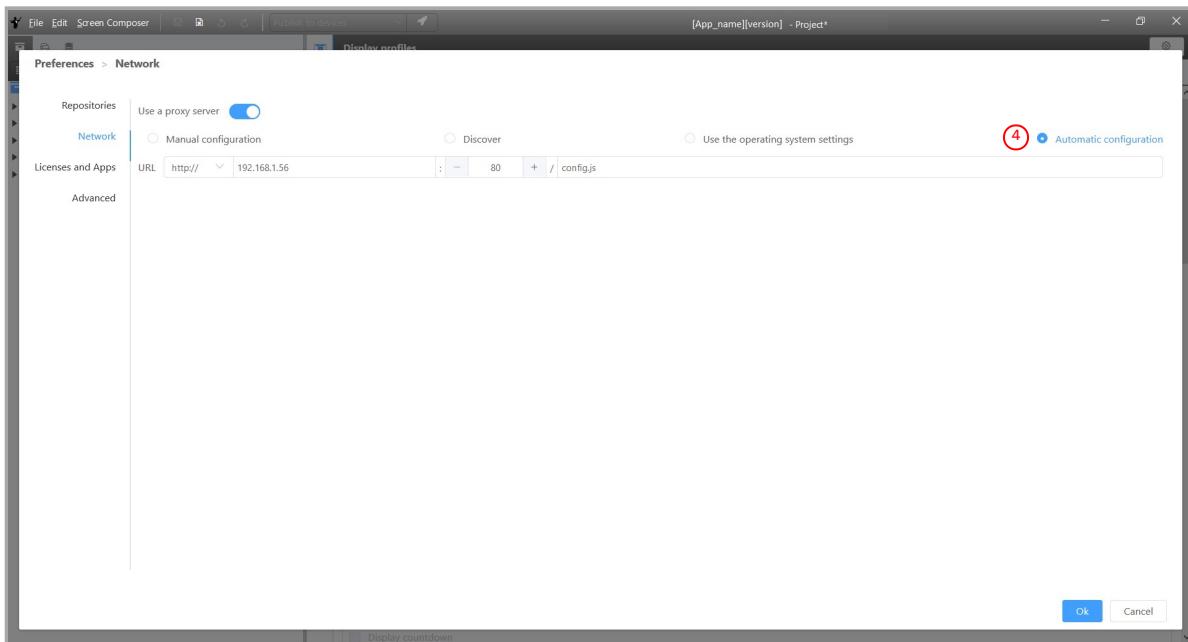
Select the Discover radio to try to detect a proxy configuration.

Proxy > Use operating system settings

Select the Use operating system settings radio to use the MS-Windows proxy configuration.

Proxy > Automatic configuration

Select the Automatic configuration radio to use a proxy configuration script. Enter here the URL to get the proxy configuration script.

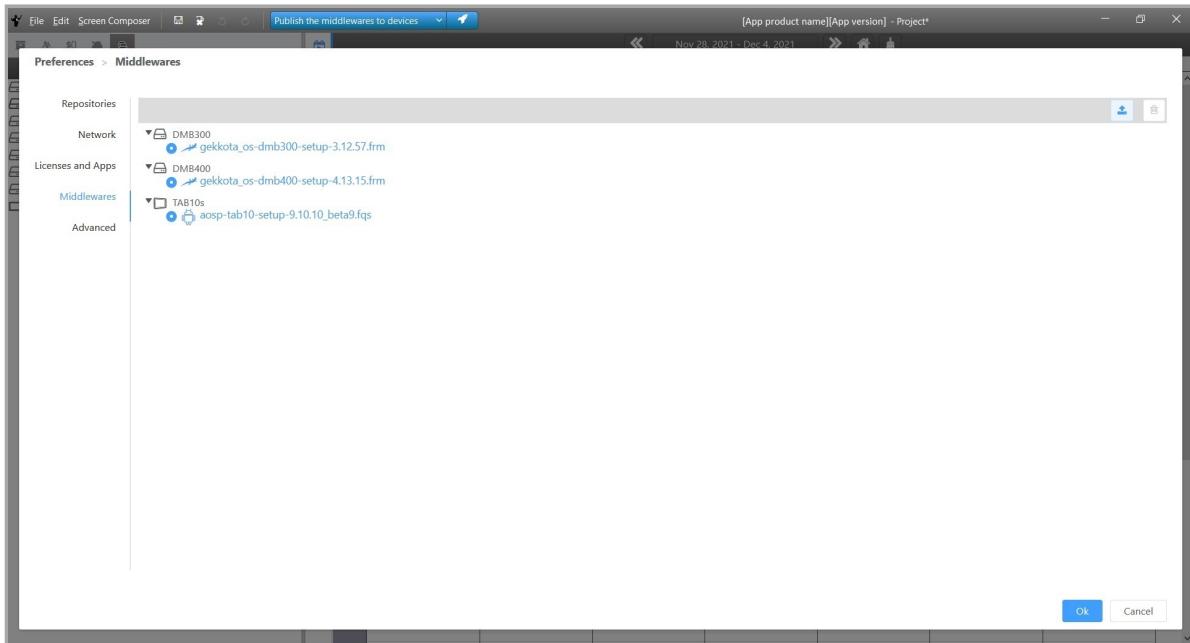


2.2.4 Screen Composer > Preferences > Middlewares

In the Preferences item of the Screen Composer menu, select the Middlewares item.

Click on the  button to choose the middlewares to import in INNES Screen Composer G4 .

 The middlewares are effectively imported in INNES Screen Composer G4 when the OK button is clicked to close the pane.



2.2.5 Screen Composer > Preferences > Licenses and Apps

License policy

There are two `core INNES` Screen Composer G4 license key delivery formats:

- a newer format based on `core INNES` Screen Composer G4 license keys pack:
- this can be a Screen Composer core for 20 devices `core` license key pack: capable of supporting up to twenty publishing tokens for publishing to devices with a single SignCom App license key,
- this can be a Screen Composer core for 5 devices `core` license key pack: capable of supporting up to five publishing tokens for publishing to devices with a single SignCom App license key,
- it can also be an additional Screen Composer core for 10 additional devices `core` license key pack: capable of supporting up to ten additional publishing tokens to publish to devices with still a single SignCom App license key, bringing the total number of supported devices to respectively thirty (twenty + ten) or fifteen (five + ten).¹
- an older format based on a single `core INNES` Screen Composer G4 license key:
- this Screen Composer core license key is capable of supporting older several SignCom App ² license keys.

¹ It is possible to install other Screen Composer core for 10 additional devices license keys packs to support more publishing tokens to publish on devices (e.g.: 20 + 10 + 10 + 10 + or 5 + 10 + 10 + 10 + ...).

² In this case, you also need as many SignCom App license keys as devices registered in your INNES Screen Composer G4 projects to have the required publishing tokens.

Installation

The INNES Screen Composer G4 core license keys have been delivered by INNES to the recipient email provided by the customer (e.g. `cico@contoso.fr`) with this kind of email body¹:

- Screen Composer core license key format:

Copy the core license key line below and paste it in.
Core license key :
`WViS1 - ffaf - wuTW - S4ab - zz/ff; PSN01200-00251 CD1; Screen Composer core`

- Screen Composer core for 20 devices license key format:

Copy the core license key line below and paste it in.
Core license key :
`ZWTxF - ff/o - w7Y1 - Gziz - ncIff ; PSN01202-00000 CD3 ; Screen Composer core for 20 devices`

- Screen Composer core for 10 additional devices license key format:

Copy the core license key line below and paste it in.
Core license key :
`MDHar - ffJS - QL7h - 194D - DJlff ; PSN01560-00000 CD4 ; Screen Composer core for 10 additional devices`

The license shown above is a fake license.

Open this email, and copy the entire line with the Screen Composer core suffix included.

- Screen Composer core license key format:

`WViS1 - ffaf - wuTW - S4ab - zz/ff; PSN01200-00251 CD1; Screen Composer core`

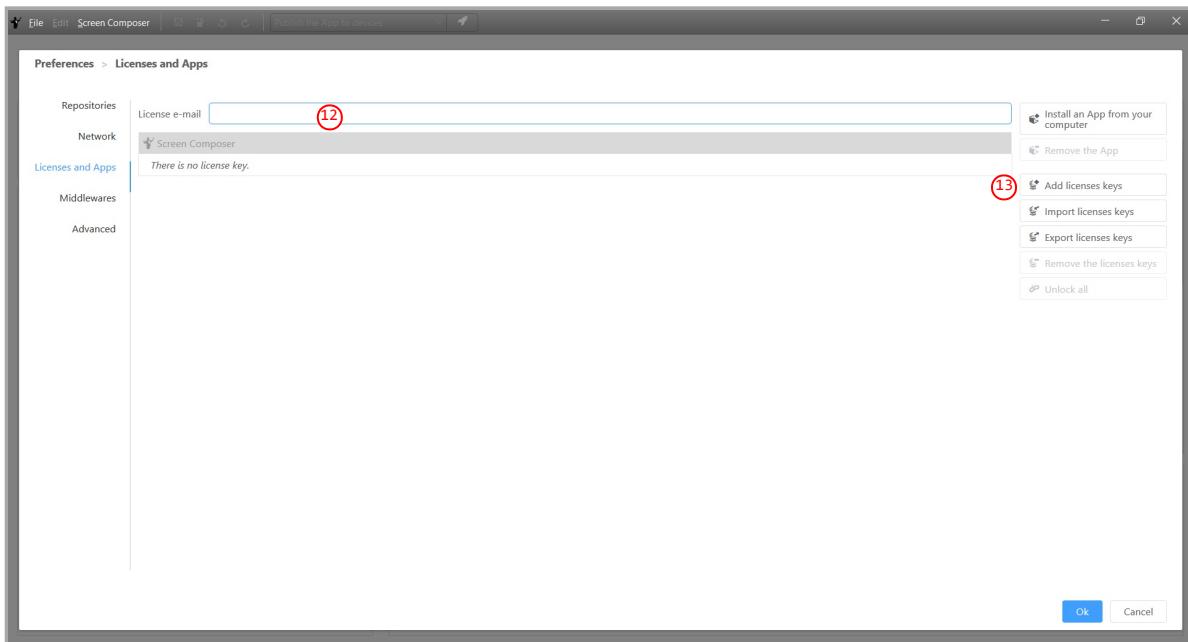
- Screen Composer core for 20 devices license key format and optional Screen Composer core for 10 additional devices license key formats:

`ZWTxF - ff/o - w7Y1 - Gziz - ncIff ; PSN01202-00000 CD3 ; Screen Composer core for 20 devices`
`MDHar - ffJS - QL7h - 194D - DJlff ; PSN01560-00000 CD4 ; Screen Composer core for 10 additional devices`

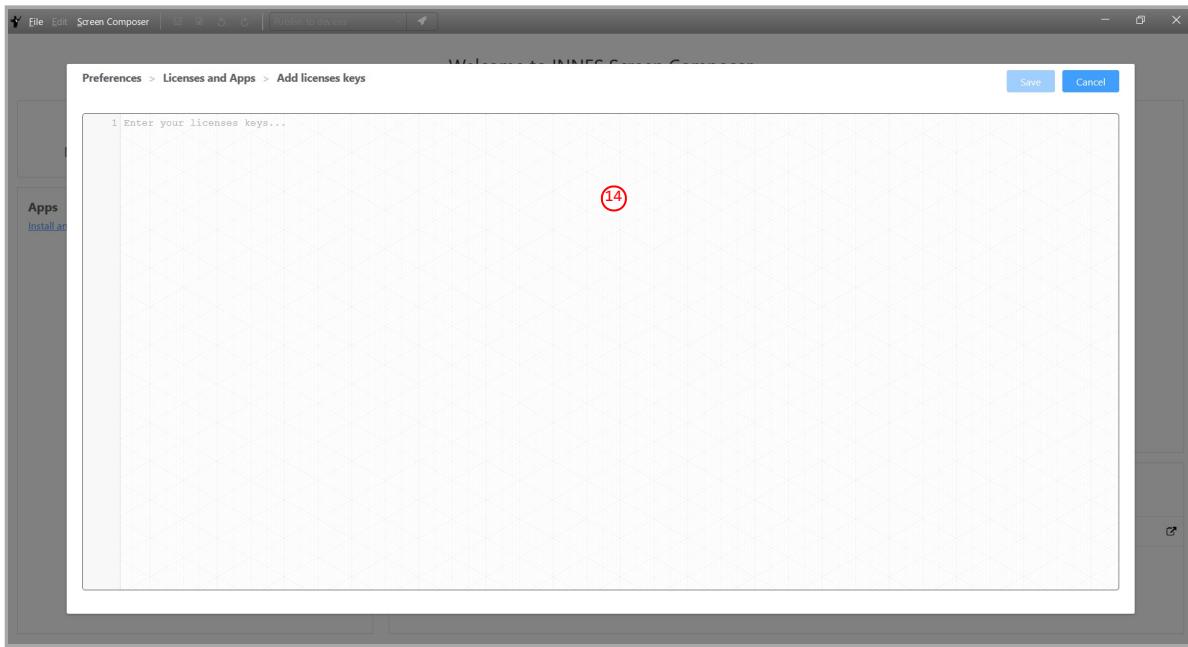
 In case you have other licenses keys linked to a specific SignCom App product, you can copy them and paste them as well. A new SignCom App part will be created in the pane and the licenses will be spread accordingly on the Screen Composer side and on the SignCom App side. For further information, refer to the chapter § [App license keys installation](#).

To access to the Licenses and Apps pane, select the Preferences item of the Screen Composer menu. To work with INNES Screen Composer G4 with this computer:

- enter the License e-mail value ^⑫ of the e-mail account used to get the licenses (e.g. `ciso@contoso.fr`),
- click on the  Add licenses keys ^⑬ button to add the Screen Composer core license key.



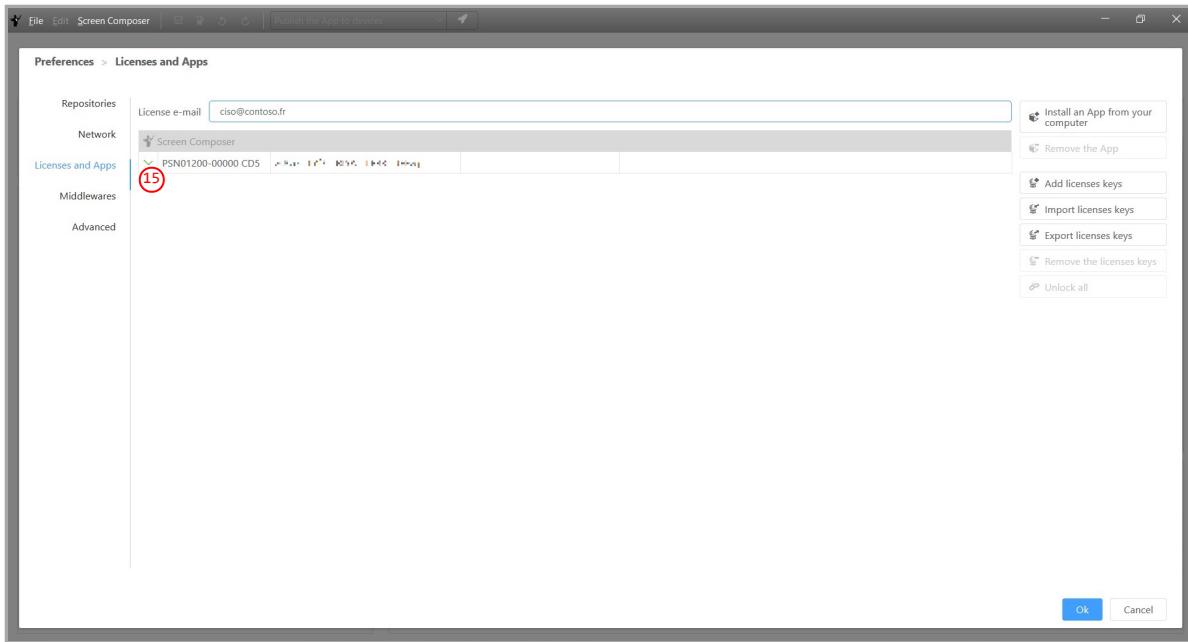
Paste ⑯ the license coming from your e-mail body.



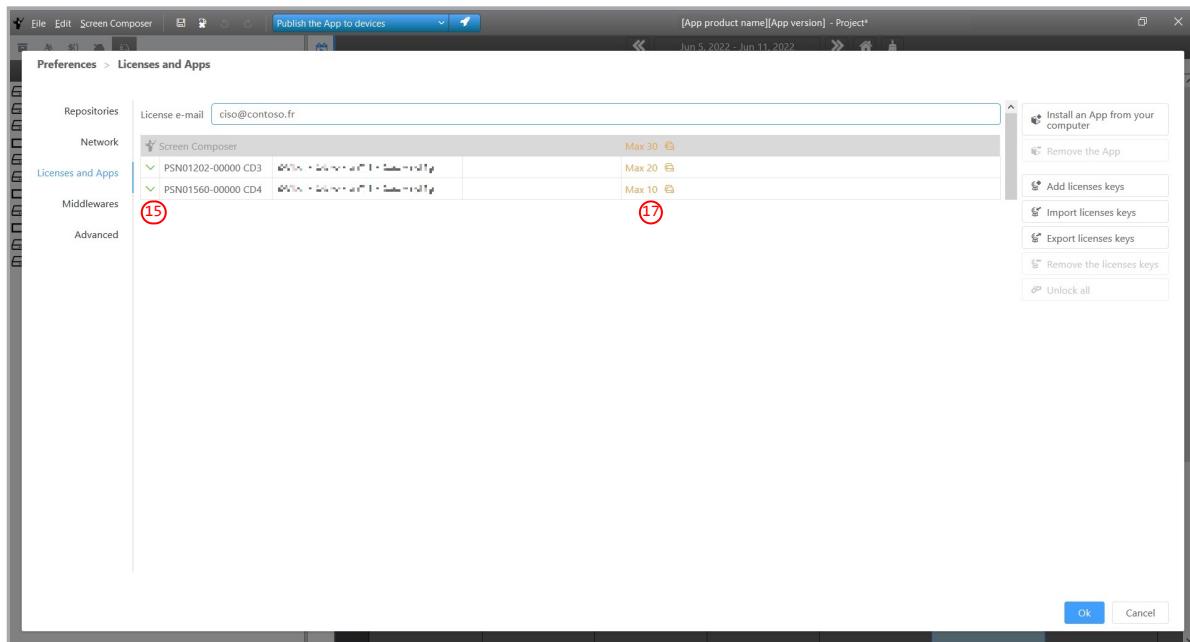
For example:

A green checkmark ⑰ is showing that the license key is valid.

- Screen Composer core license key format:



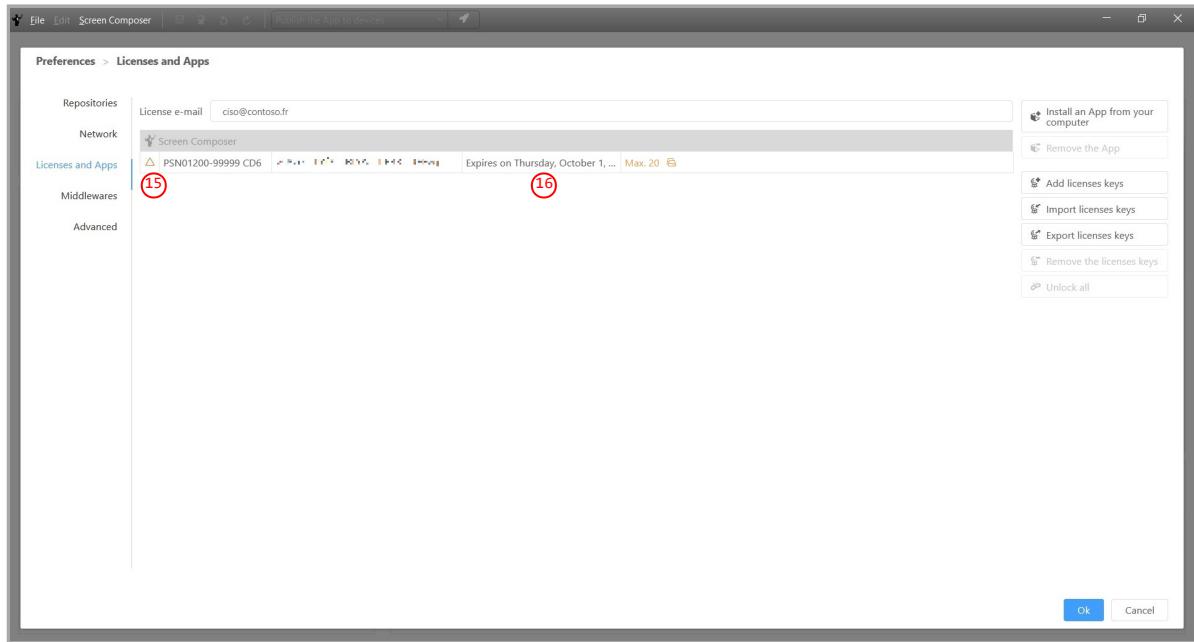
- Screen Composer core for 20 devices and optional Screen Composer core for 10 additional devices license key formats:



The max. number of devices (in the example: 20 + 10 devices) supported in your project appears in this column (in the example: Max 30 = Max 20 + Max 10).

Demo licenses

In case demo license key, the expiration date **⑯** and the max. of supported devices appear (e.g.: *Expires on Sunday, May 1, 2022, Max 5 devices*).



2.2.5.1 App installation

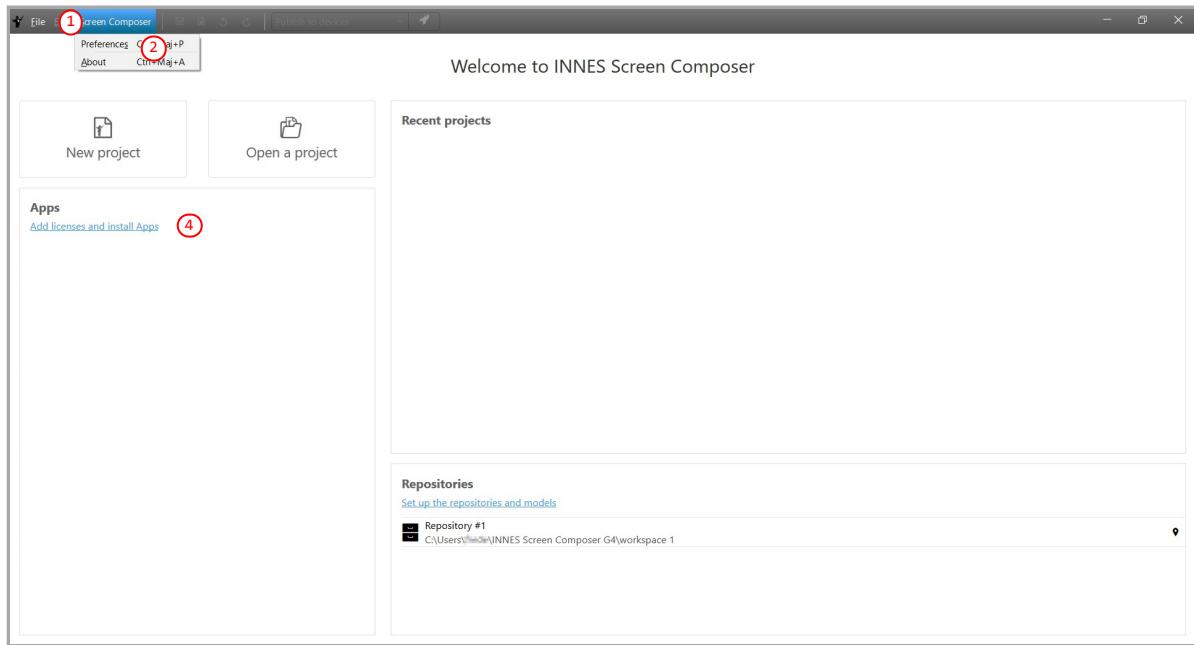
App download and installation

Before creating a new INNES Screen Composer G4 project, you need get the last available SignCom App version available for download from the [INNES Web site](#).

The SignCom App file matches this pattern: `signcom-screen-composer-addin-2.<yy>.<zz>.appi`.

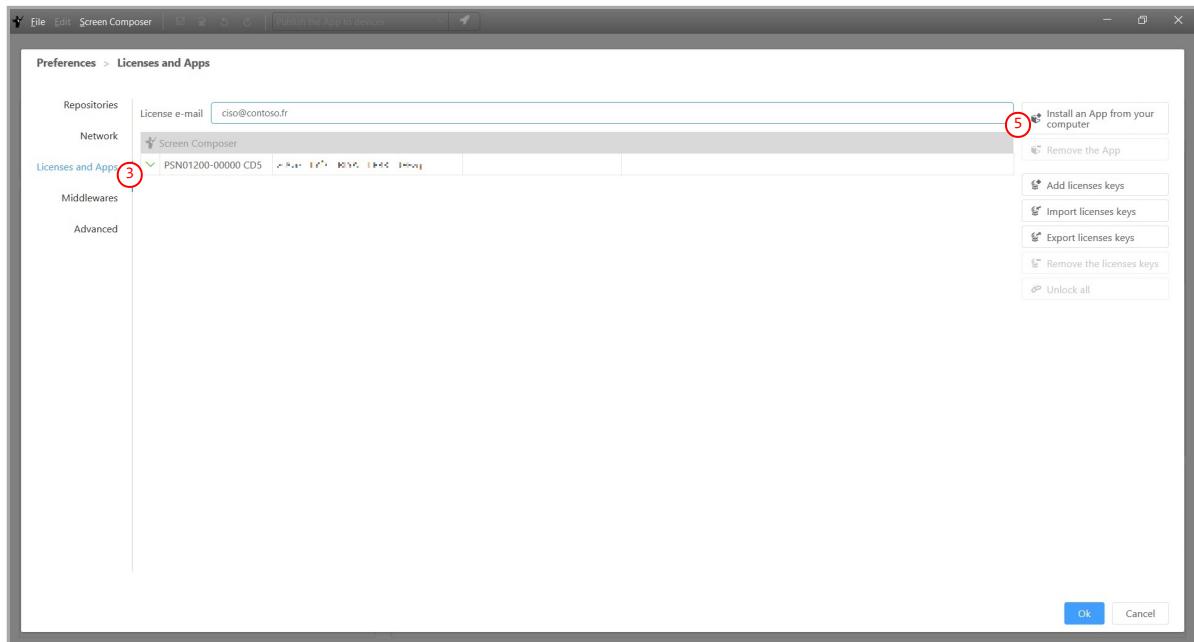
From the welcome screen:

- either click on the Add licenses and install Apps **(4)** blue link,
- or click on the Preferences **(2)** item of the Screen Composer **(1)** menu then select the Licenses and Apps **(3)** item.

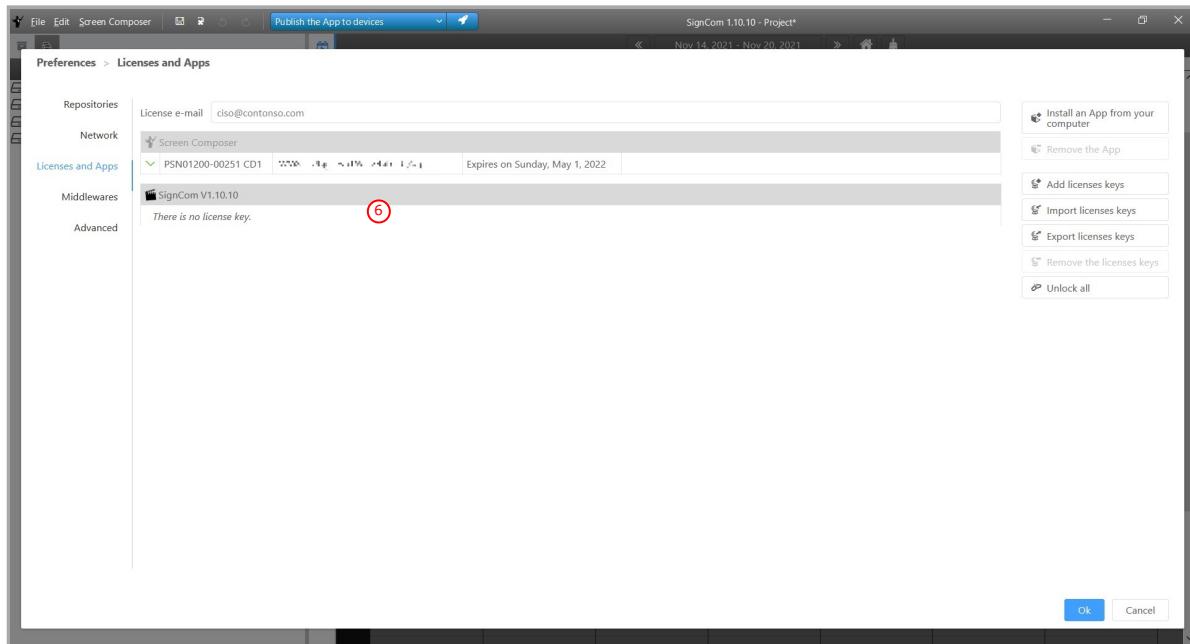


Click on the Install an App from your computer **(5)** and select the appropriate SignCom App file:

- Filename pattern: `signcom-screen_composer-addin-<xx>.<yy>.<zz>.appi`.



The SignCom App (6) appears in the pane.



To remove the App, select the App and click on the Remove the App button.

It is not possible to remove an App product which is currently used by the opened project.

Once installed, the SignCom App is stored automatically in the following directory `C:\Users\<username>\AppData\Roaming\INNES Screen Composer G4\innes\appis\`.

2.2.5.2 App license keys installation

License policy

In this version, two SignCom App license key formats are supported:

- either one SignCom App license key per device,
- or one SignCom App license key common for all your devices¹.

¹ That requires to have a specific Screen Composer core license keys able to manage key license pack for twenty devices and key license pack for five devices.

Each SignCom App license key is associated to :

- a user email address (e.g. *ciso@contoso.fr*)
- a SignCom App product (e.g. SignCom for Screen Composer)

This one for one SignCom App license key per device:

- ⚠ You must have as much SignCom App license keys as you have devices in your project.
- ⚠ A SignCom App license key cannot be used twice for two different devices in two different projects.
- ⚠ A SignCom App license key value can be paired to only one device, which is identified by its `MAC Id`. The pairing is done at the publication.
- ☞ Once the SignCom App licenses keys are installed, they are kept permanently in `INNES Screen Composer G4` and are visible for every projects.

This other one for one SignCom App license key common for all you devices:

- ⚠ You have one SignCom App license keys common for all your devices.

Installation

The SignCom App license keys have been delivered by INNES to the user email address provided by the customer (e.g. *ciso@contoso.fr*) with this kind of user email body¹:

This one for one SignCom App license key per device:

```
- Copy all the Apps license key lines below (at once) and paste them in.  
Apps License Keys (Qty = 3) :  
Enter the appropriate licenses keys for your App product  
HhBo - om6gl - px7Jp - 8itiV - sViff ; PSN01480-00000 CD8 ; SignCom for Screen Composer  
kpjle - 5tew - x7uq - ubsee - 1zffff ; PSN01480-00001 CD5 ; SignCom for Screen Composer  
tk5cK - efdCF - lyf1 - 8os1 - iuyff ; PSN01480-00002 CD2 ; SignCom for Screen Composer
```

This other one for one SignCom App license key common for all you devices:

```
- Copy all the Apps license key lines below (at once) and paste them in.  
Apps License Keys (Qty = 1) :  
Enter the appropriate licenses keys for your App product  
HhBo - om6gl - px7Jp - 8itiV - sViff ;
```

The license shown above are fake license.

- ☞ When importing the SignCom App license keys, a suffix is showing which App product, the license keys are given for.
- ☞ The license keys used for `INNES Screen Composer G3 / Playzilla` are suitable for `INNES Screen Composer G4 / SignCom`.

Copy the entire line with the `SignCom for Screen Composer` suffix included.

This one for one SignCom App license key per device:

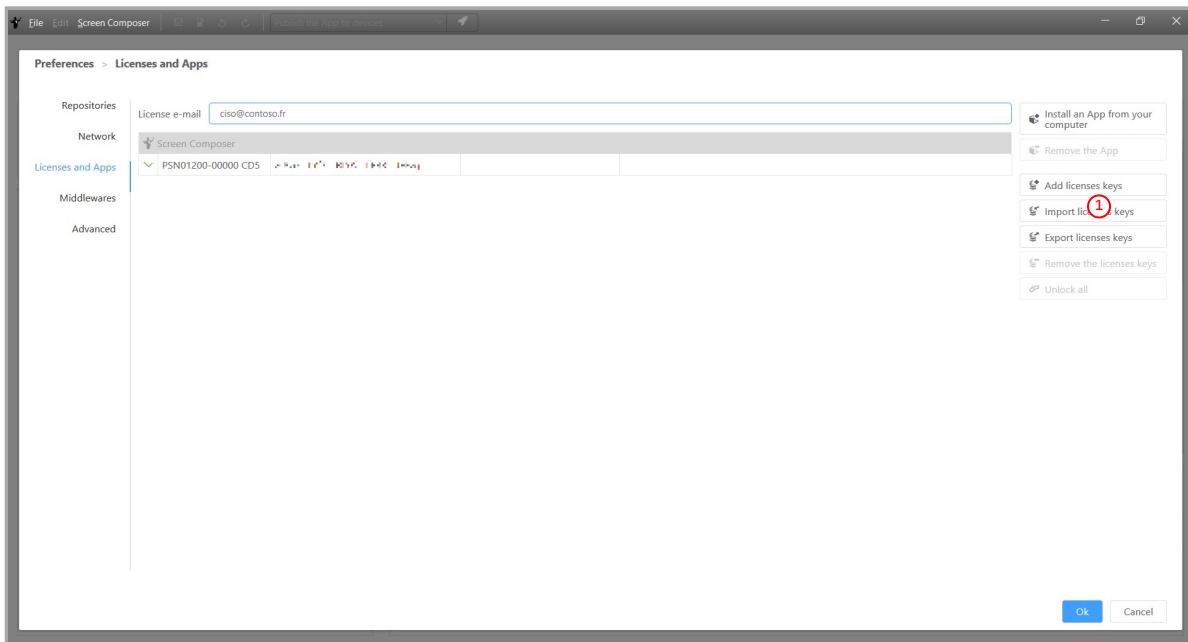
```
HhBo - om6gl - px7Jp - 8itiV - sViff ; PSN01480-00000 CD8 ; SignCom for Screen Composer  
kpjle - 5tew - x7uq - ubsee - 1zffff ; PSN01480-00001 CD5 ; SignCom for Screen Composer  
tk5cK - efdCF - lyf1 - 8os1 - iuyff ; PSN01480-00002 CD2 ; SignCom for Screen Composer
```

The other one for SignCom App license key common for all you devices:

```
HhBo - om6gl - px7Jp - 8itiV - sViff ; PSN01480-00000 CD8 ; SignCom for Screen Composer
```

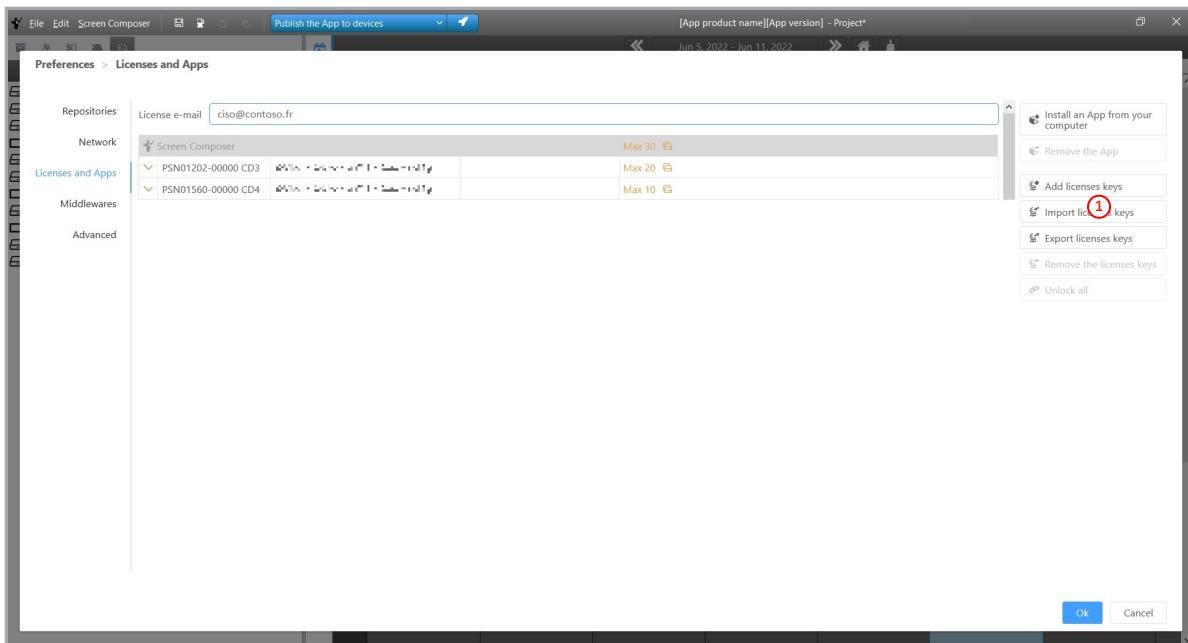
This one for one SignCom App license key per device:

Click on the  Add licenses keys ① button to add the SignCom App licenses keys.



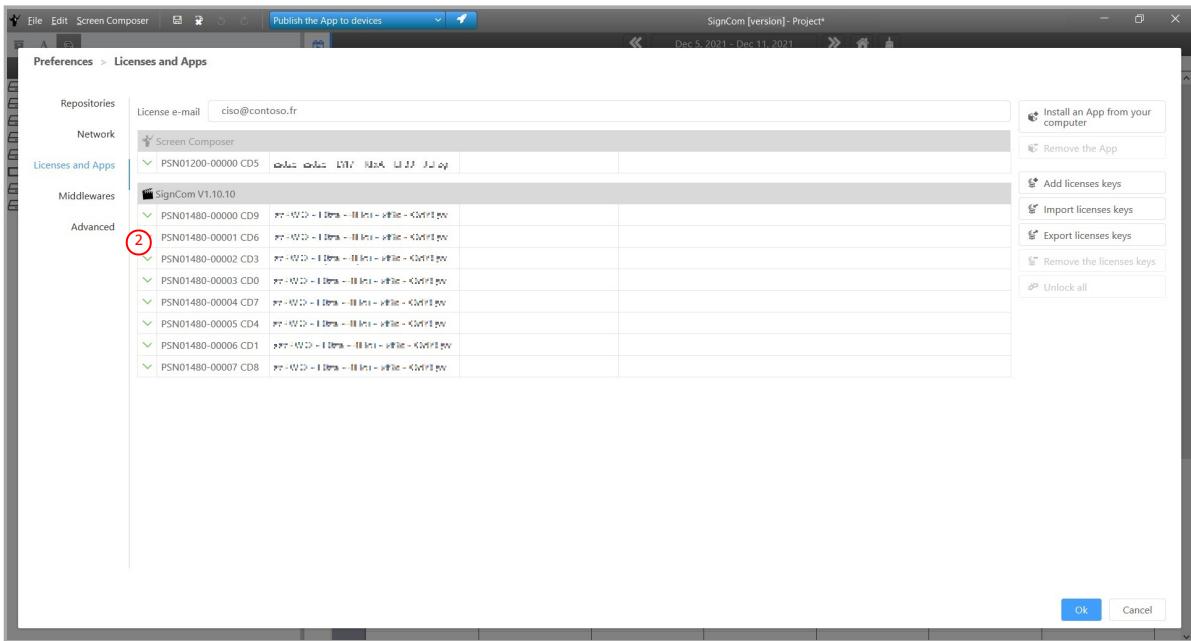
This other one for one SignCom App license key common for all you devices:

Click on the Add licenses keys **①** button to add the SignCom App license key.

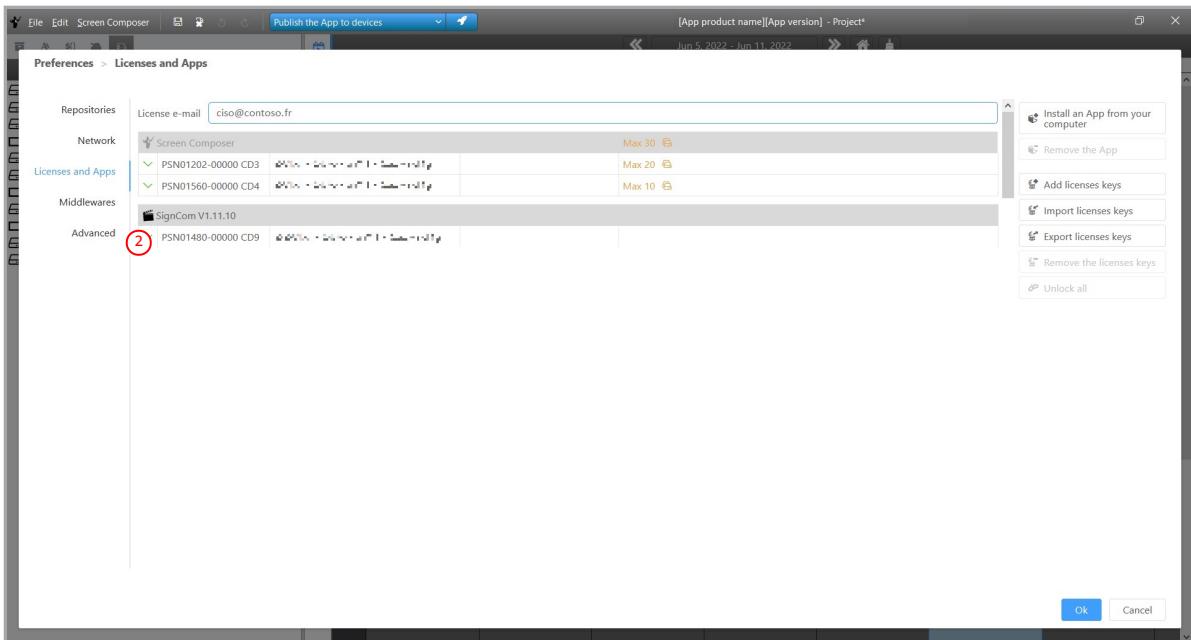


A green checkmark (2) is showing that the license keys are valid.

This one for one SignCom App license key per device:



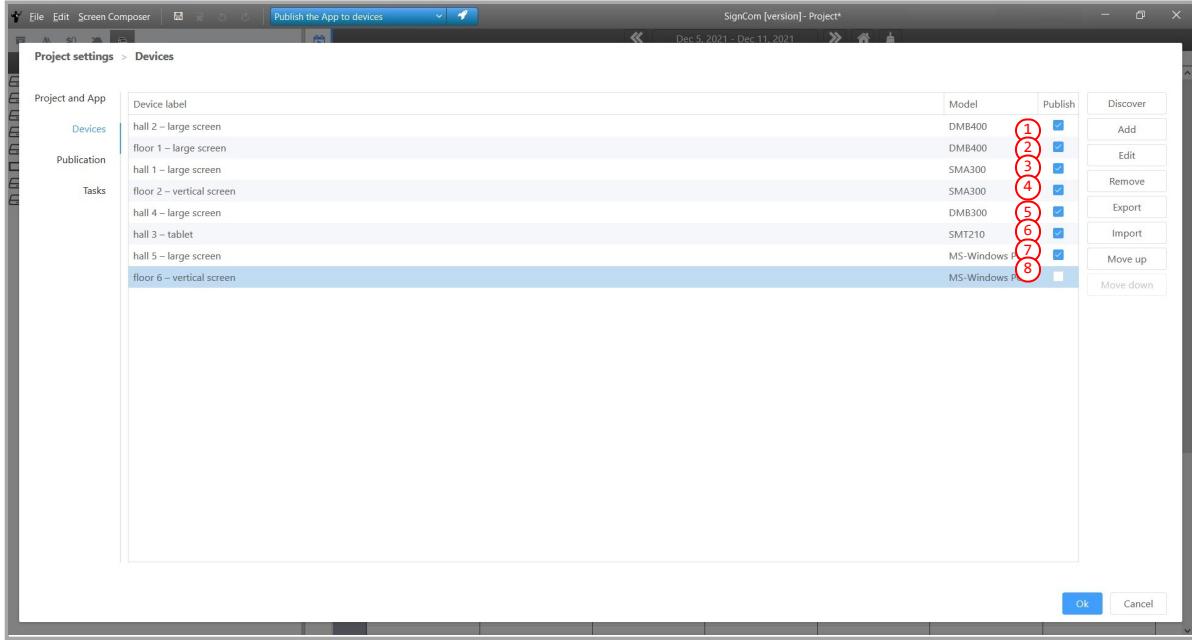
This other one for one SignCom App license key common for all your devices:



Publication and App license key pairing

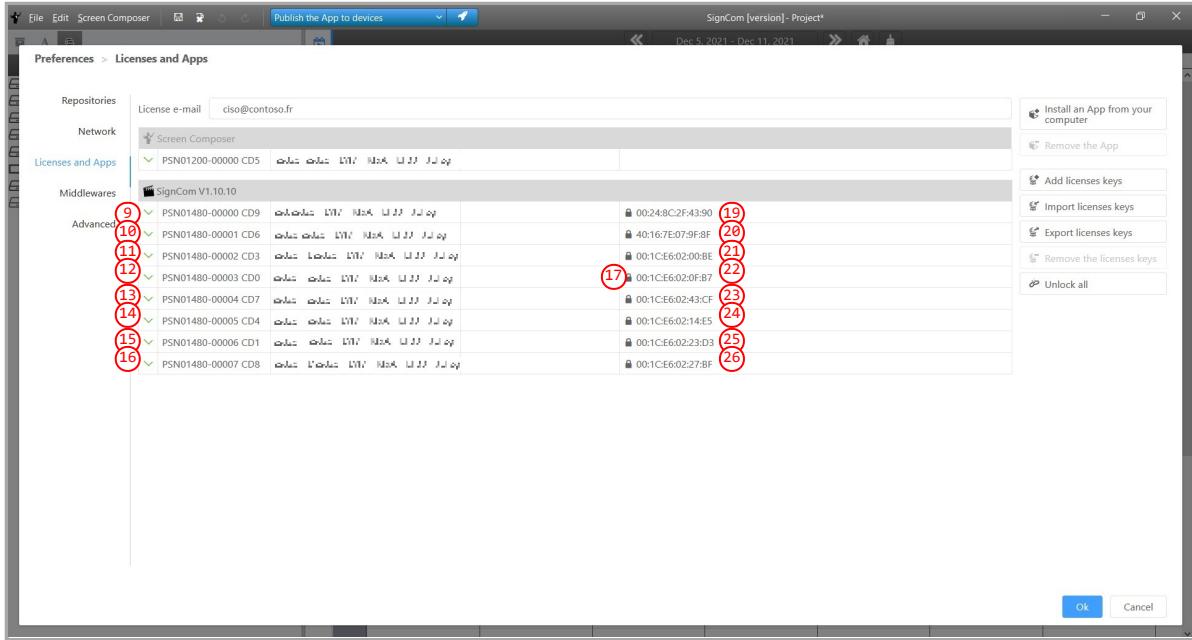
This one for one SignCom App license key per device:

To publish successfully, the number of devices added to your project (e.g. ① to ⑧) must be equal to (or be lower than) the number of SignCom App license keys to pair (e.g. 8 SignCom App licenses keys from ⑨ to ⑯).

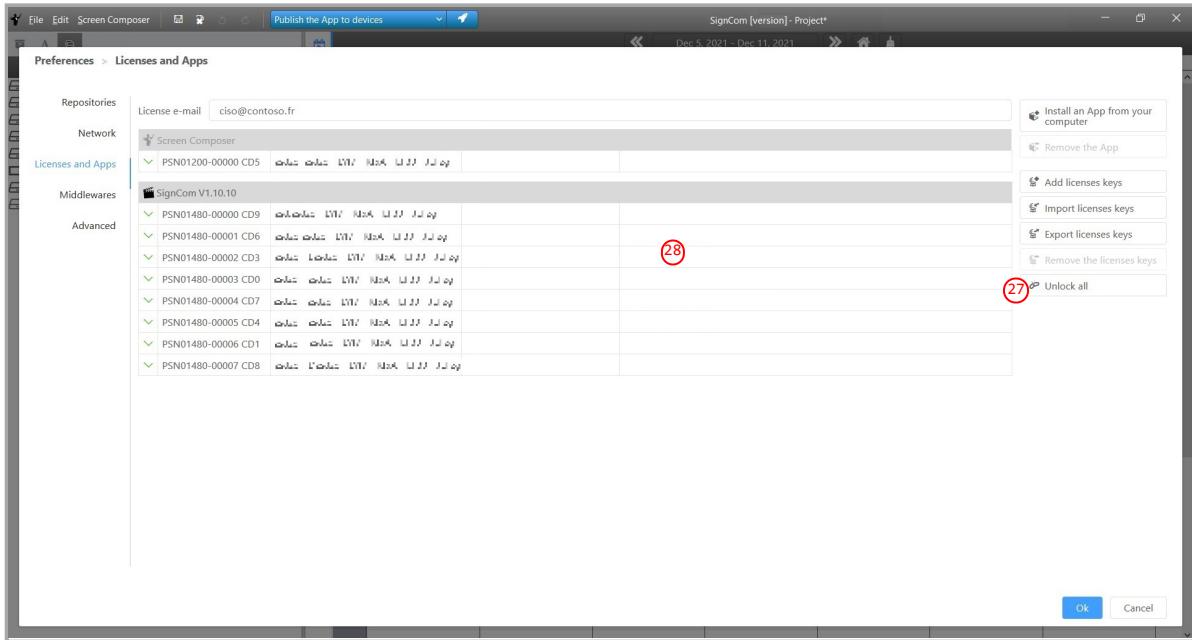


When the publication is completed, INNES Screen Composer G4 keeps the pairing status of the SignCom App license keys with the project's devices MAC Id.

The SignCom App license key's pairing status is identified by a padlock ⑰ meaning that each SignCom App license key is paired to a specific device.



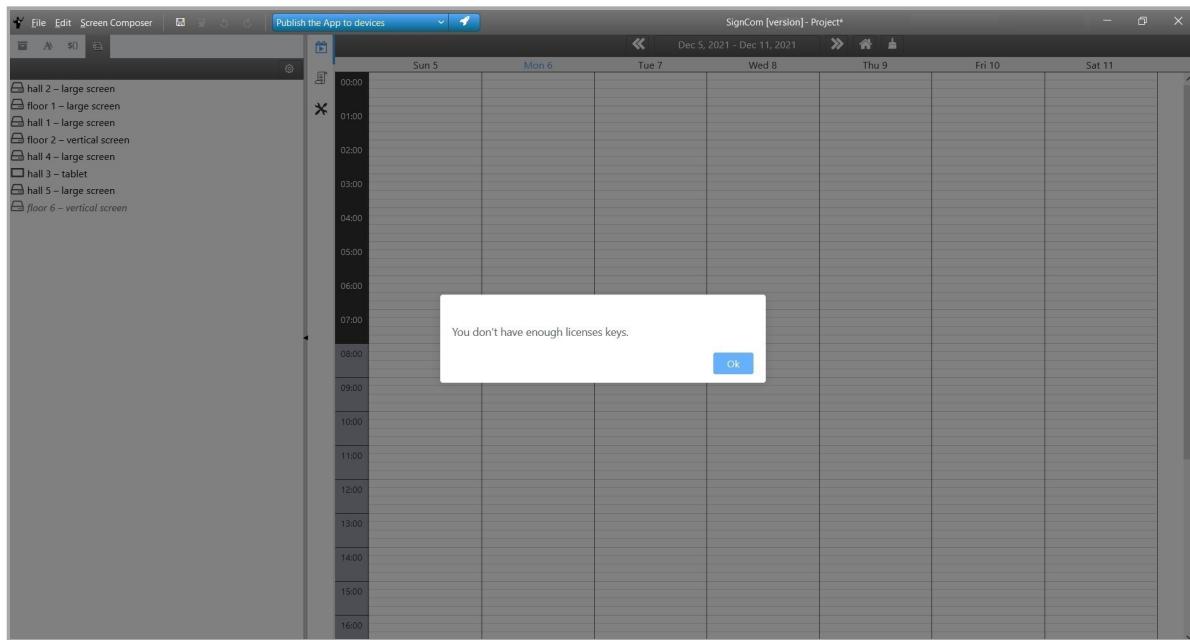
In case, you have added an unexpected device to your project after having published on it by mistake, an SignCom App license key value is used for nothing. To work around, remove this device from your project and click on the *Unlock all*   button to forget  the overall SignCom App license key pairing configuration then publish again.



This other one for one SignCom App license key common for all you devices:

To publish successfully, the number of devices added to your project (e.g.  to  must be equal to (or be lower than) the number of device supported by your Screen Composer core license keys (e.g. Max 30  = Max 20  + Max 10 ).

In case the number of SignCom App license keys is not sufficient or your Screen Composer core licence key does not support enough devices, an information message is raised when the publication is launched.



To work around,

- add the appropriate number of SignCom App license keys missing to your project or,
- add a Screen Composer core licence key for 10 additional devices (if ever already a Screen Composer core licence key for twenty devices is already installed),
- remove definitively the excess devices from your project.

2.2.6 Screen Composer > Preferences > Advanced

In the Preferences item of the Screen Composer menu, select the Advanced item.

Set the toggle button to the right (blue color) to Open the last project as soon as a INNES Screen Composer G4 is launched.



Set the toggle button to the left (grey color) to display the welcome pane instead of to Open the last project as soon as a INNES Screen Composer G4 is launched.



2.2.7 File > Project settings > Devices

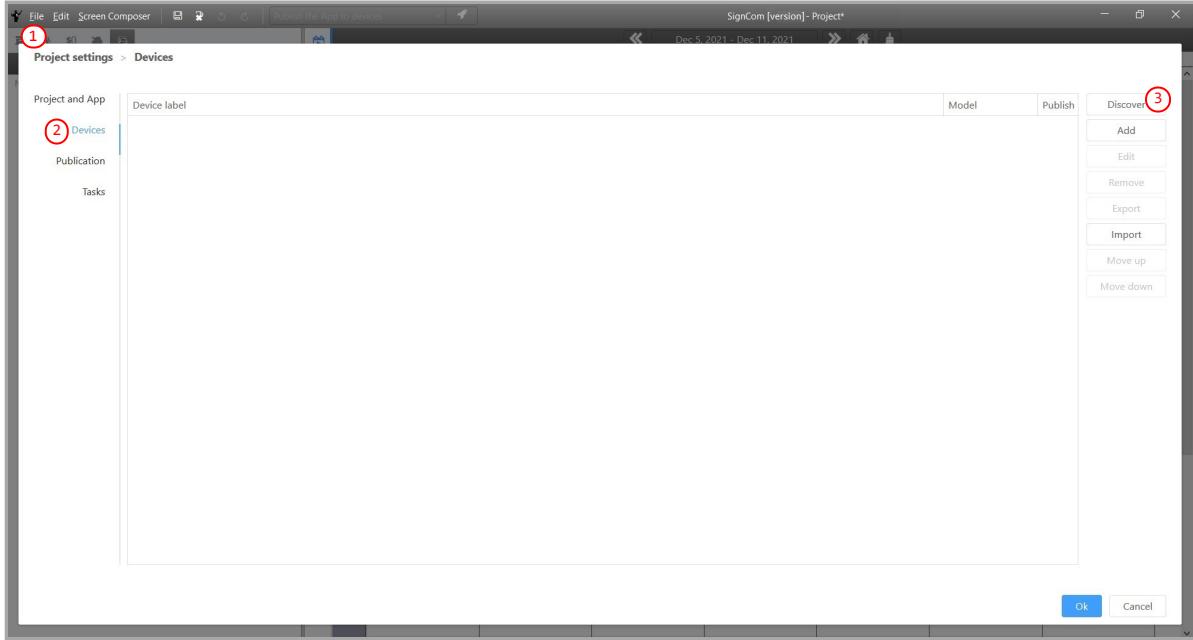
Before launching a devices discovery (UPnP) on your local network, check that the devices are properly configured to work with INNES Screen Composer G4 and to work with the SignCom App.

For further information about Gekkota 4 devices, refer to the chapter § [Gekkota devices configuration for INNES Screen Composer G4](#).

To add devices to your project, click on the `Project settings` item of the `File` (1) menu, then select the `Devices` (2) item.

Click on the `Discover` (3) button to discover by UPnP the devices which are alive on the local network.

INNES Screen Composer G4 can detect Gekkota 4, Gekkota 3 and Qeedji System AOSP 9 devices.



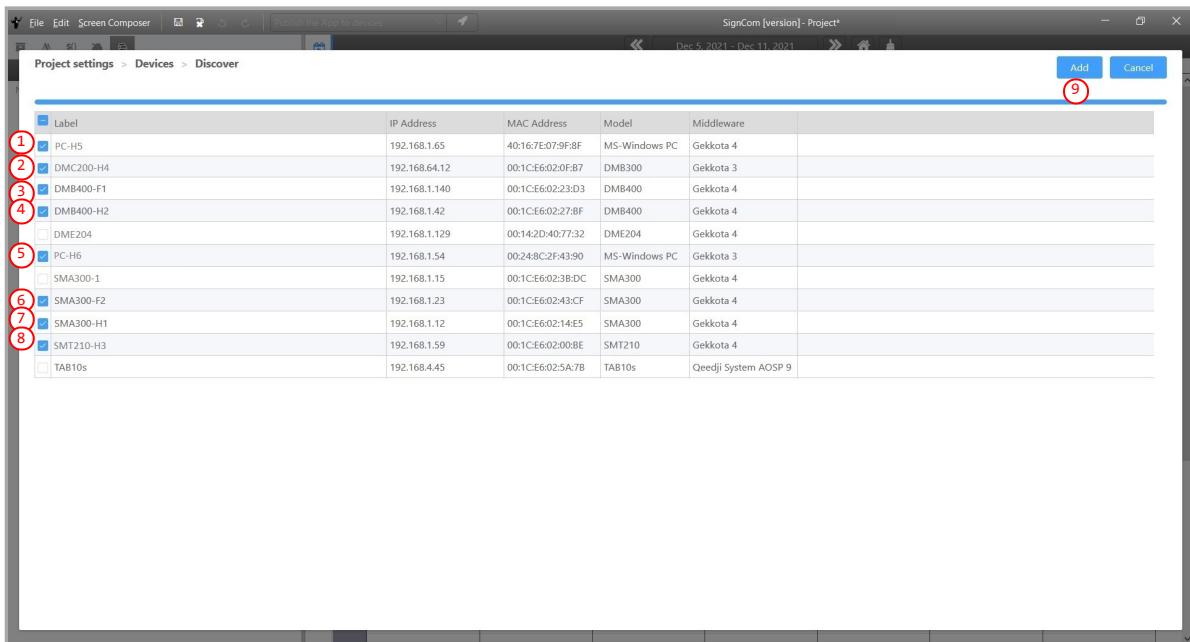
A list of the devices is detected.

The screenshot shows the 'Discover' dialog box within the 'Project settings > Devices' window. It lists 15 devices with their details:

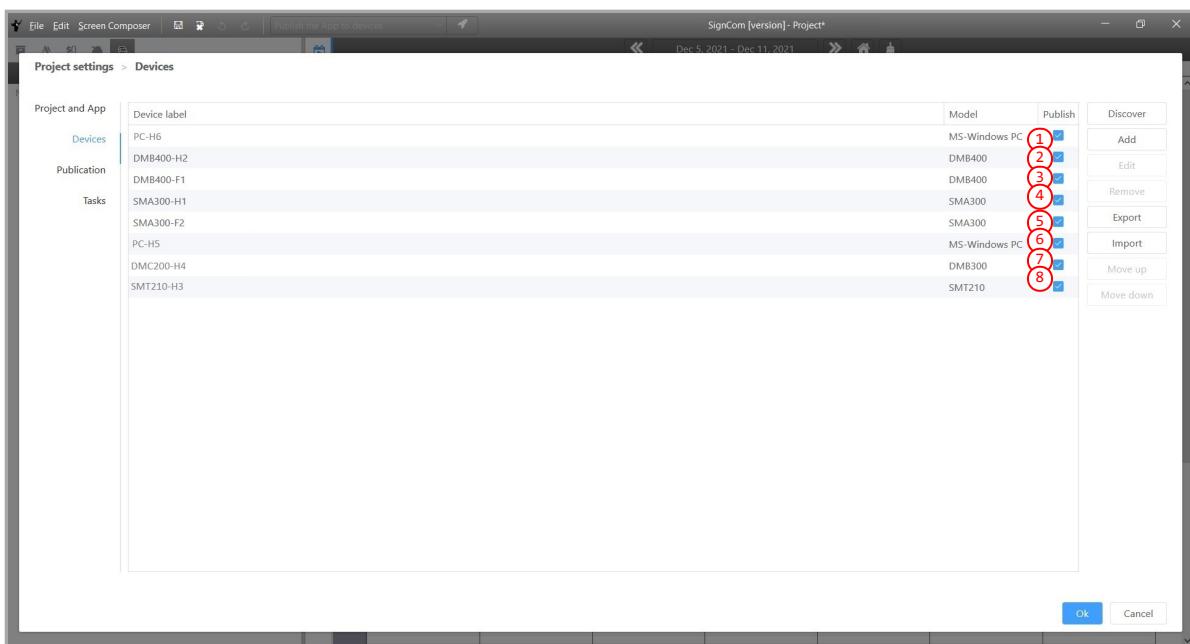
Label	IP Address	MAC Address	Model	Middleware
PC-H5	192.168.1.65	40:16:7E:07:9F:8F	MS-Windows PC	Gekkota 4
DMC200-H4	192.168.64.12	00:1CE6:02:0F:B7	DMB300	Gekkota 3
DMB400-F1	192.168.1.140	00:1CE6:02:23:D3	DMB400	Gekkota 4
DMB400-H2	192.168.1.42	00:1CE6:02:27:BF	DMB400	Gekkota 4
DME204	192.168.1.129	00:14:D0:40:77:32	DME204	Gekkota 4
PC-H6	192.168.1.54	00:24:8C:2F:43:90	MS-Windows PC	Gekkota 3
SMA300	192.168.1.15	00:1CE6:02:3B:DC	SMA300	Gekkota 4
SMA300-F2	192.168.1.23	00:1CE6:02:43:CF	SMA300	Gekkota 4
SMA300-H1	192.168.1.12	00:1CE6:02:14:E5	SMA300	Gekkota 4
SMT210-1	192.168.1.96	00:1CE6:02:00:C1	SMT210	Gekkota 4
SMT210-2	192.168.64.210	00:1CE6:02:00:C3	SMT210	Gekkota 4
TAB10FRD1s	192.168.4.45	00:1CE6:02:5A:7B	TAB10s	Qeedji System AOSP 9

The SignCom App is not compatible with AOSP devices.

In the list of the devices detected, select the appropriate devices ①, ②, ③, ④, ⑤, ⑥, ⑦, ⑧ to be added to your project. Then click on the Add ⑨ button.



Once the devices are added to your project, select the one ①, ②, ③, ④, ⑤, ⑥, ⑦, ⑧ applicant for the publication.



When the devices are detected thanks to the UPnP discovery, the default `admin/admin` login credentials and the default `http://` scheme value may be not consistent. Double check these parameters each time you add a new device to the project.

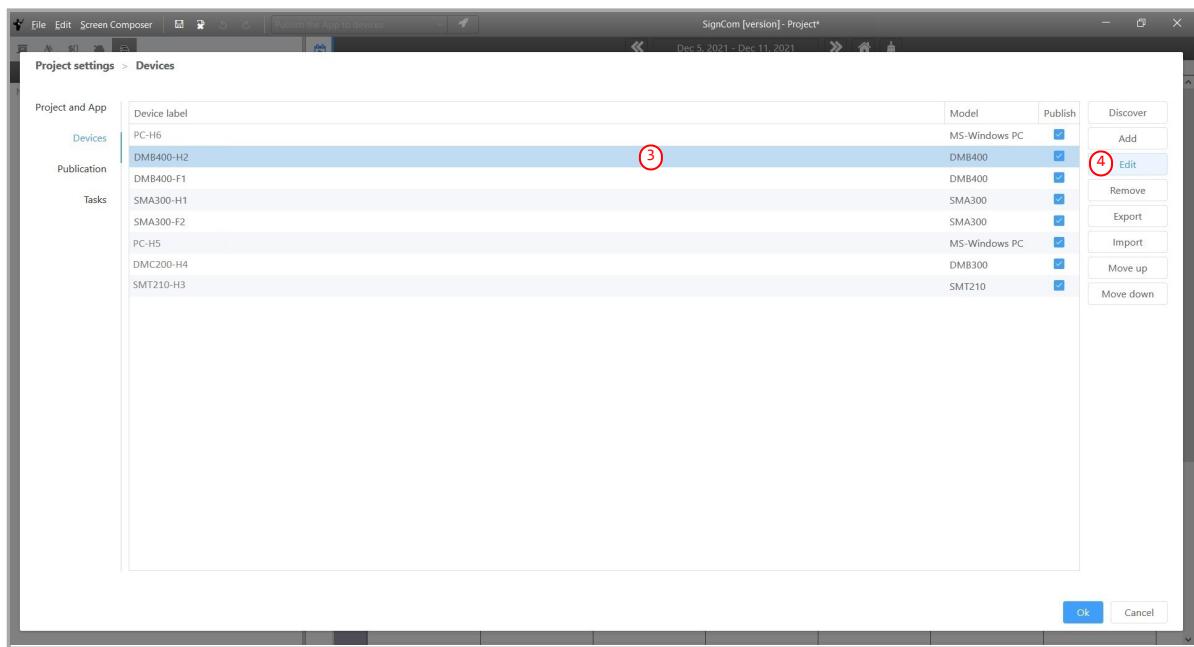
⚠ The login credentials per each device cannot be automatically detected and are set by default to `admin / admin` when a device is added to the project. If required, modify the login credentials with consistent values.

⚠ The device only available with the `https://` scheme cannot be detected automatically.

⚠ The port value to access to this the device is automatically detected.

⚠ Once the devices are added to your project with their appropriate scheme and login credentials, you can export the device configuration into a `.csv` file with the Export ① button. With the Import ② button, it is possible afterwards to import the devices configuration.

To change the login credentials, the scheme value or the port value to access to the Web server of the device with INNES Screen Composer G4 , select the appropriate device in the list **③** an click on the **>Edit** **④** button.

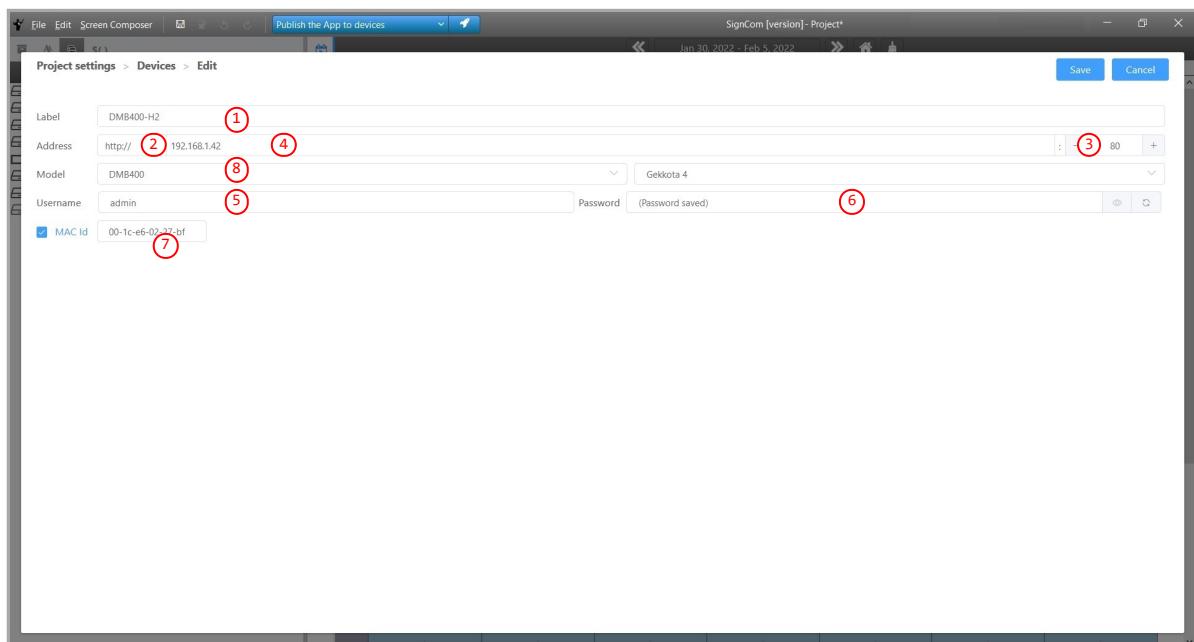


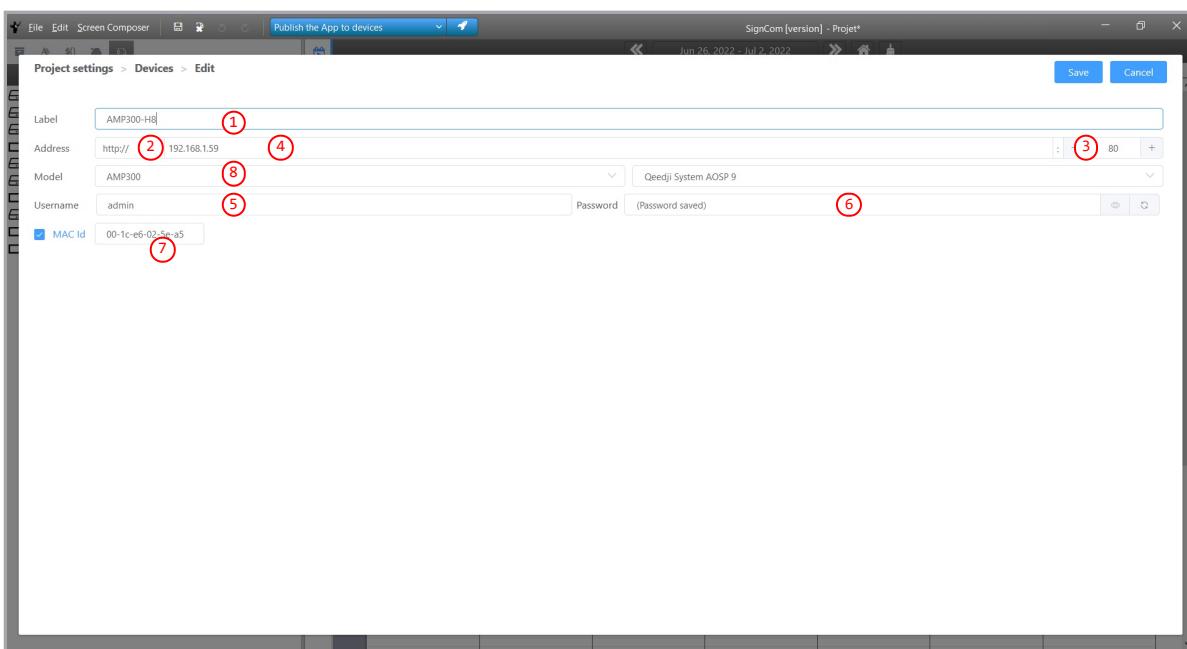
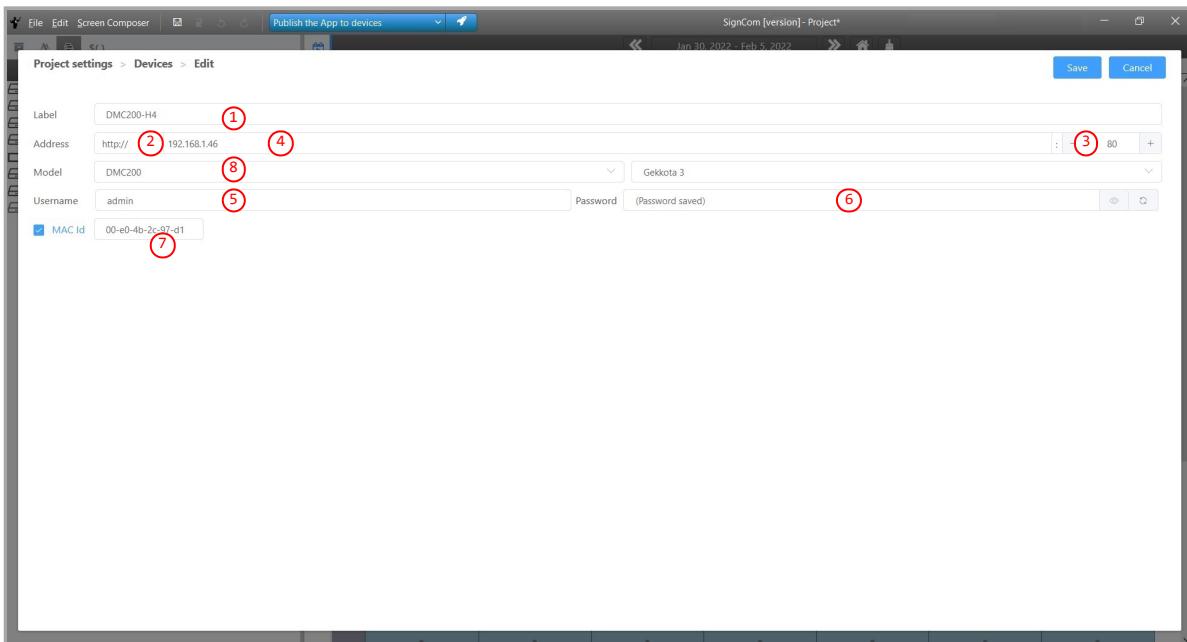
For each Gekkota devices, you can watch or modify:

- the device label **①** in INNES Screen Composer G4 : the default label value given here is the device's hostname value,
- the device scheme **②** on which the device is available (*http://* or *https://*),
- the device port **③**,
- the device IP address **④**,
- the Username **⑤** and password **⑥**: login credentials to access to the device WebDAV server,
- the MAC Id **⑦**: MAC identifier for the license¹,
- the model and OS **⑧**.

¹ Using an invalid MAC Id for a device should lead to the display of a black content when SignCom App is running.

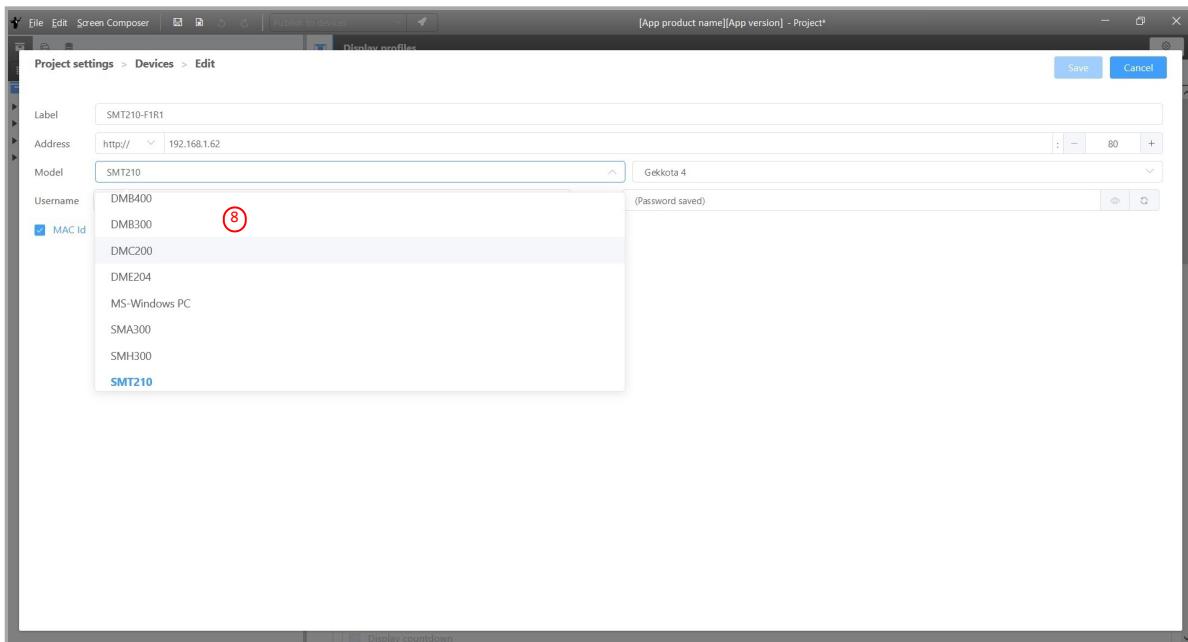
² For Qeedji System AOSP devices, the MAC ID must match the MAC address value of the WLAN interface. For Gekkota 3 and Gekkota 4 devices, the MAC Id must match the MAC address value of the LAN interface.





For Gekkota 3 and Gekkota 4 devices, you can change the model (8) among the values here after:

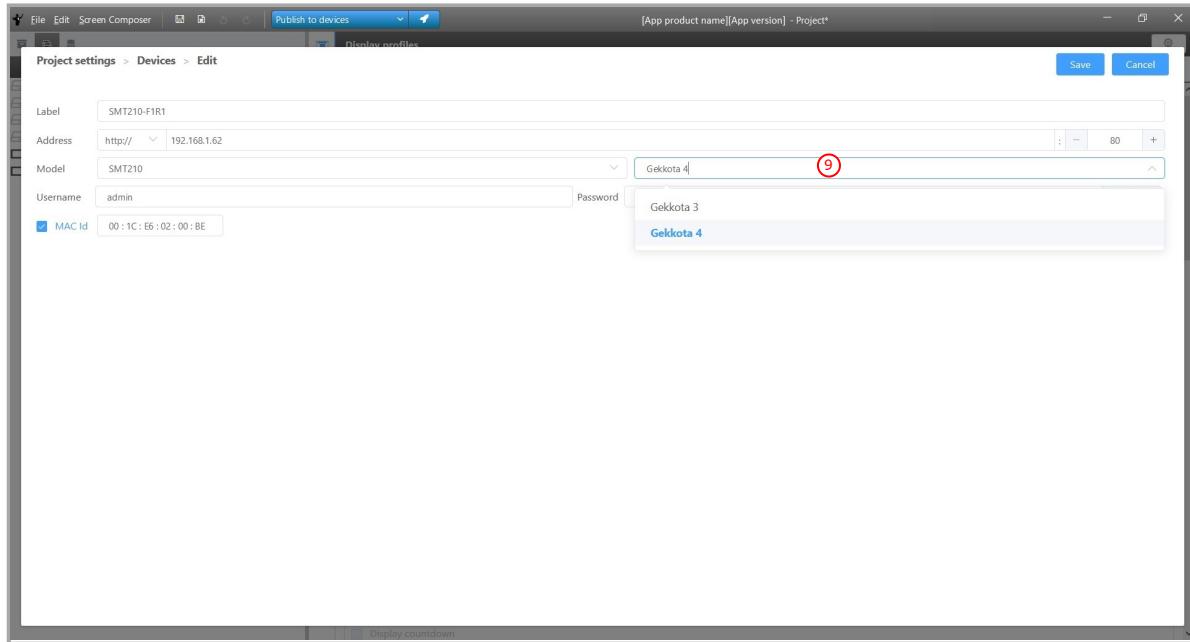
- SMT210,
- DMB400,
- SMA300,
- SMH300,
- DME204,
- MS-Windows PC,
- DMB300,
- DMC200.



For SMA300, SMT210 and MS-Windows PC, after having chosen the device model, choose the device OS ⑨ among the suggested value:

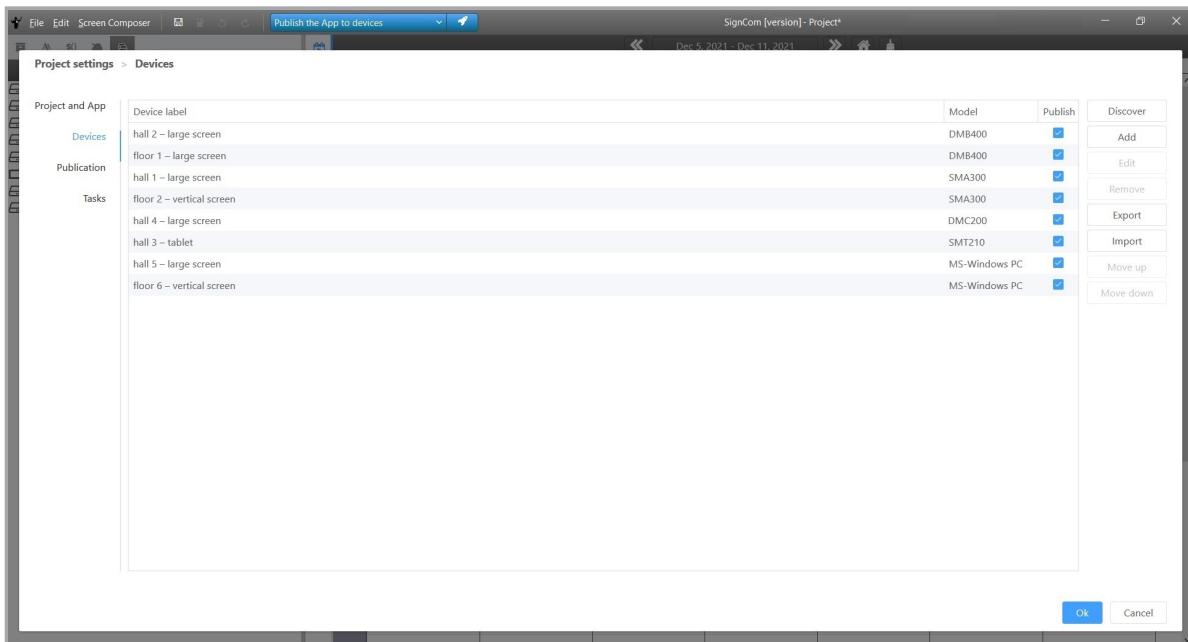
- Gekkota 4,
- Gekkota 3³.

³ Some SignCom feature like the Enforce on media with pagination a page progression with a default page duration option is not working when it is running on Gekkota 3 devices.



In this user manual, the label for each device have been modified to define the location of the device in the building.

Device model	Device OS	Hostname received from devices during a discovery	Device label for its localization
DMB400	Gekkota 4	DMB400-H2	hall 2 – large screen
DMB400	Gekkota 4	DMB400-F1	floor 1 – large screen
SMA300	Gekkota 4	SMA300-H1	hall 1 – large screen
SMA300	Gekkota 4	SMA300-F2	floor 2 – vertical screen
SMT210	Gekkota 4	SMT210-H3	hall 3 – tablet
DMB300	Gekkota 4	DMC200_H4	hall 4 – large screen
MS-Windows PC	Gekkota 4	PC1_H5	hall 5 – large screen
MS-Windows PC	Gekkota 3	PC2_H6	hall 6 – vertical screen
AMP300	AOSP	AMP300-H8	hall 8 – large screen
TAB10s	AOSP	TAB10-H7	hall 7 – tablet



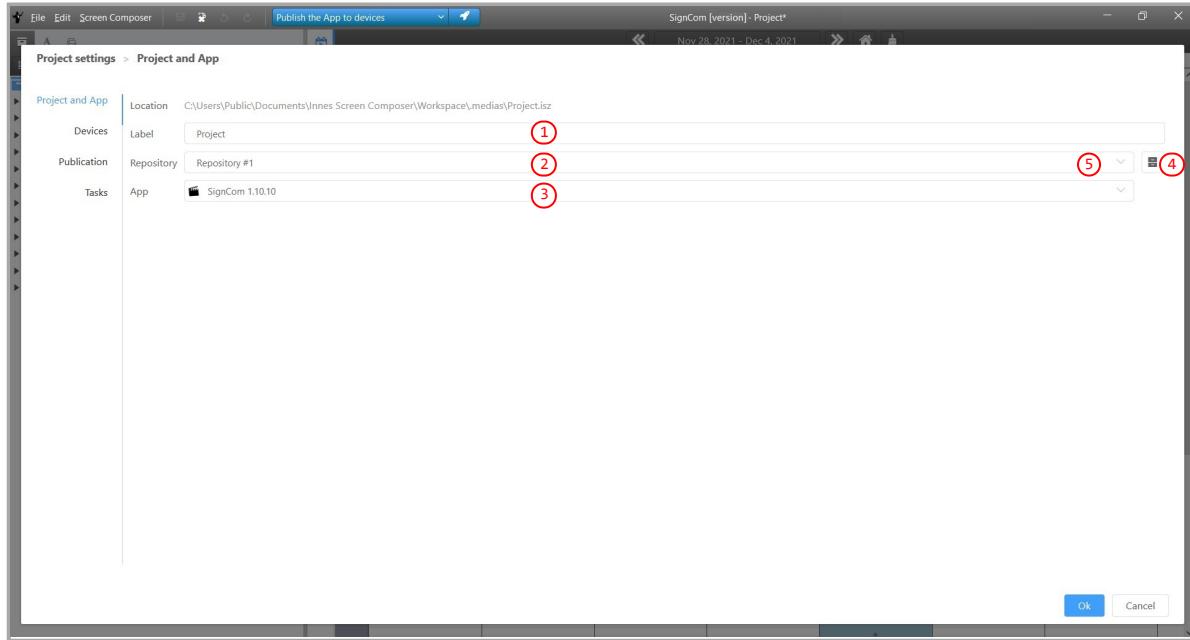
2.2.8 File > Project settings > Project and App

In the Project settings item of the File menu, select the Project and App item. The panel allow to show:

- the **①** label of your project,
- the **②** repository name used by your project,
- the **③** SignCom App name and SignCom App version.

To create a new project repository , click on the **④** Edit repositories button. For further information, refer to the chapter § [Screen Composer > Preferences > Repositories](#).

To change the repository for your project, click on the **⑤** drop down button.



When you have just made a `repository#` change, it is advised to check whether the used image, video or css set in the SignCom form are still present in the new repository. Else the publication may not work. To work around, it is advised to close the project and open the project again. In that case, all the inconsistent file will be removed automatically from the SignCom form. A red stamp will show where to set back a consistent image, video or css when required to have a successful publication.

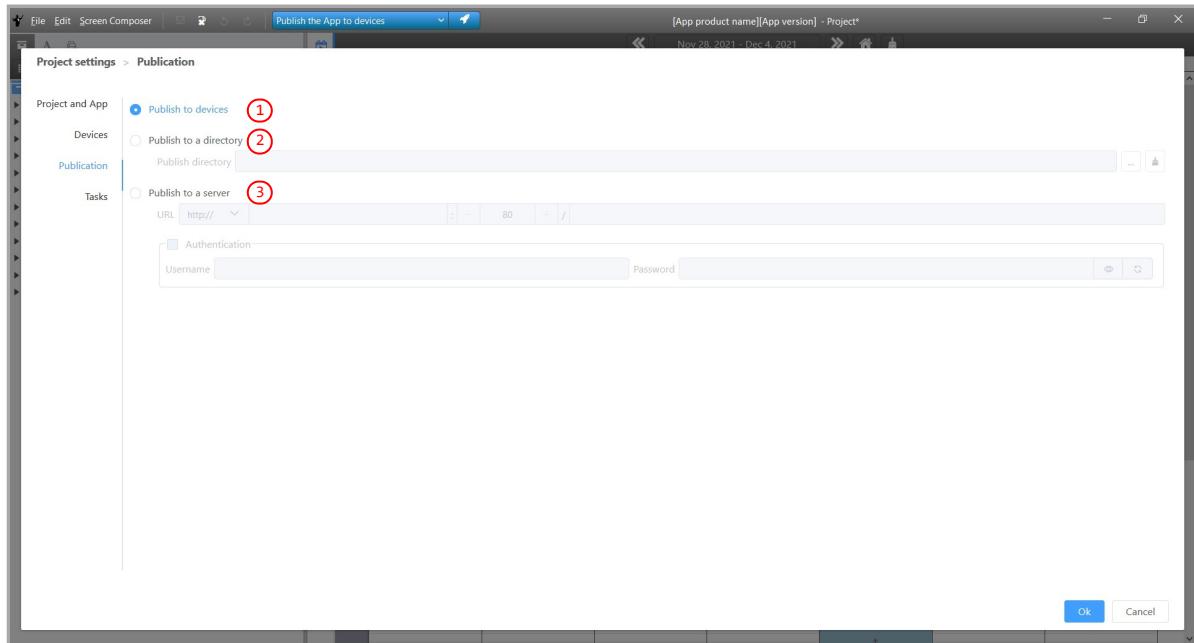
2.2.9 File > Project settings > Publication

In the Project settings item of the File menu, select the Publication item. The panel allows to choose among select several publication ways:

- Publish to devices ①,
- Publish to a directory ②,
- Publish to a server ③.

Publish the App to devices

Select the Publish to devices ① publication mode (default mode) to choose to publish the SignCom App on the devices added to the project and checked as applicant for the publication.



Activating the Publish to devices ① publication mode allows also to access to three values for the blue button at the top of the interface:

- Publish the App to devices ④: when this value is selected, clicking on the button allows to publish immediately the SignCom App to the devices.



- Publish a purge to devices ⑤: when this value is selected, clicking on the button allows to remove immediately the SignCom App from the devices. A information message No content should be then displayed on the devices.

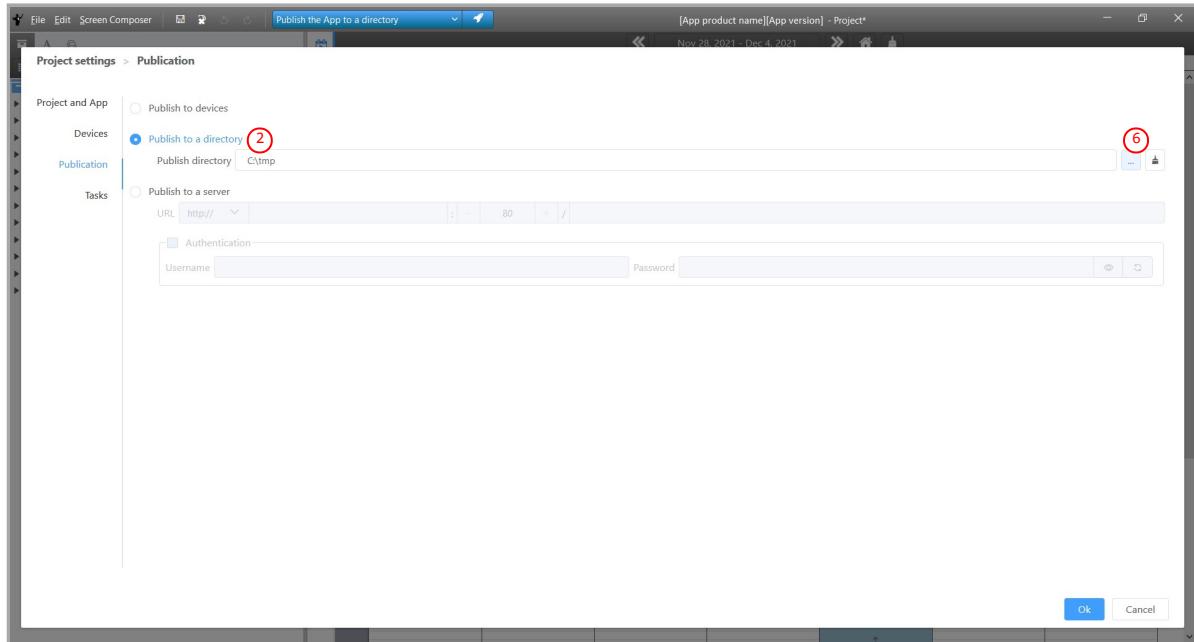


- Publish the middlewares to devices ⑥: when this value is selected, clicking on the button allows to send all the middlewares file to the devices so that the device middleware version can be updated.



Publish the App to a directory

Select the `Publish to a directory` (2) publication mode to choose to publish the App on a local directory of your computer. Then select a target directory with the file picker (6) button.



It allows also to access to a specific `Publish to a directory` (7) button. when this value is selected, clicking on the  button allows to publish immediately the SignCom App on this directory.



When the `Publish the App to a directory` and `Publish the middlewares to a directory` buttons value are available, the `Publish a purge to devices` and the `Publish an App to devices` button values are not available.

Once having clicked on the `Publish the App to a directory`, the SignCom App is copied on the directory with the appropriate manifests file with the pattern `manifest.<MAC>.xml`, with one manifest per device, making that the content is still suitable for the different devices.

App format example:

```
app/  
manifest.00-1c-e6-02-00-be.xml  
manifest.00-1c-e6-02-50-c6.xml
```

This App content can be for example copied entirely on a USB mass storage which can be then inserted in the USB port of the different devices. Each device will load automatically the appropriate part of the App.

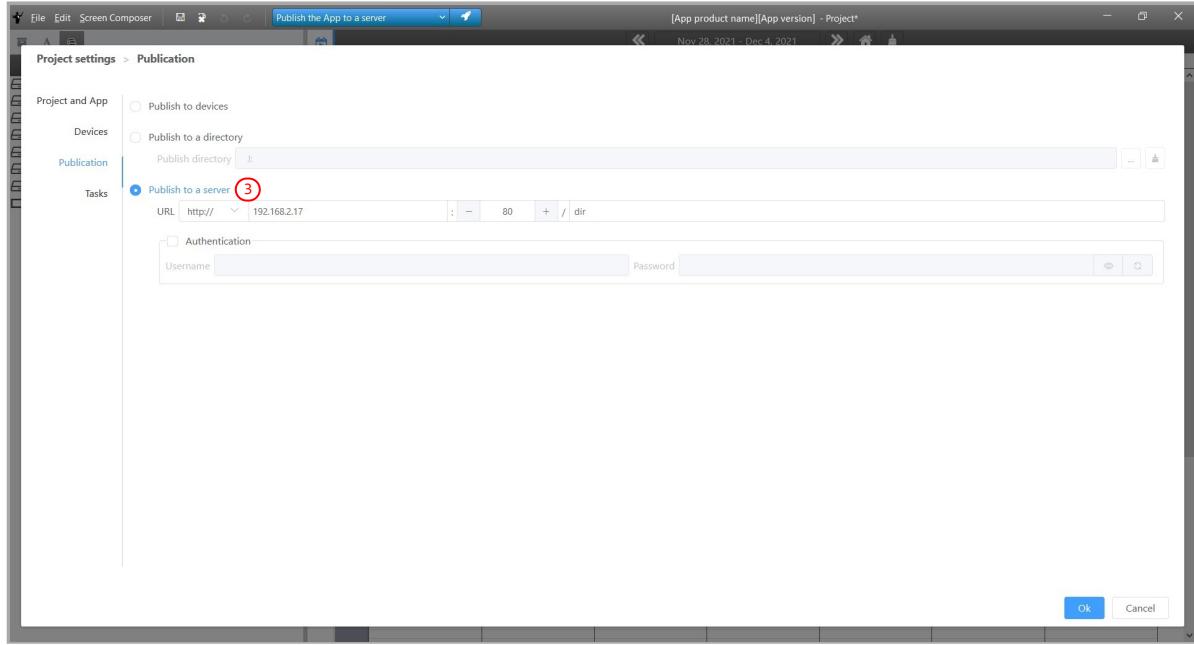
- To support the USB mass storage insertion with this `manifest.<MAC>.xml` format, the Gekkota 4 devices may require to have the identification method set to `MAC`. Do the same for `manifest.<UUID>.xml` or `manifest.<Hostname>.xml`.
- An additional `manifest.xml` is generated to support Gekkota 3 devices. For these devices, when using USB storage device to upgrade their content, it is not recommended to use variable in the App.

Publish the App to a server

Selecting the **Publish to a server** (3) publication mode allows to publish the SignCom App to a remote HTTP/WebDAV server.

Fill the required parameters to connect to the HTTP/WebDAV server:

- IP address,
- port,
- directory path,
- login credentials' username,
- login credentials' password.



When the **Publish the App to a server** (8) button value is selected, clicking on the allows to immediately publish the App content on the specified WebDAV server directory.

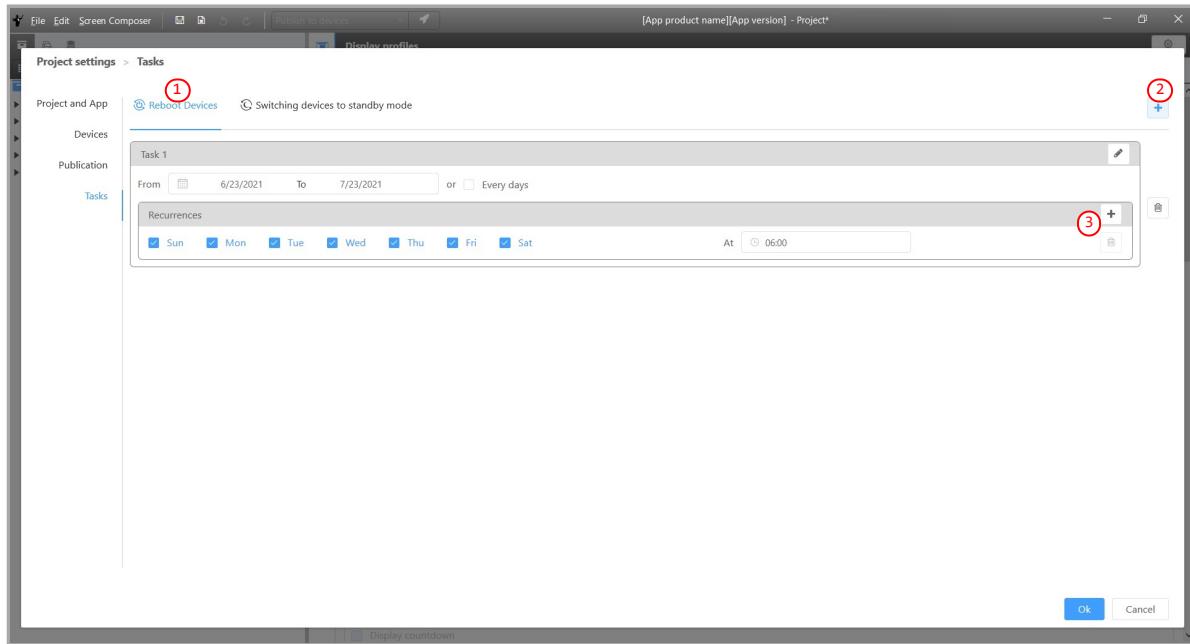


When the **Publish the App to a server** button is available, the **Publish a purge to devices** button, the **Publish an App to devices** button and the **Publish an App to a directory** button are not available anymore.

2.2.10 File > Project settings > Tasks

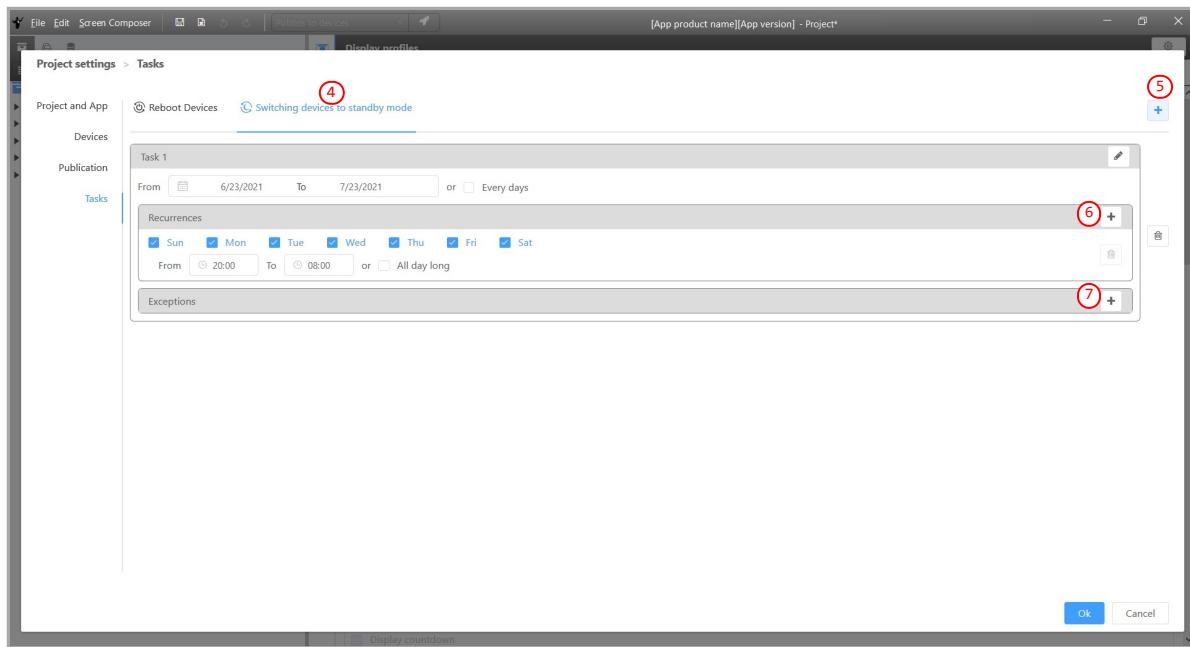
To create a planification to restart the devices registered in your project, select the `Project settings` item of the `File` menu then select the `Tasks` item. Then click on the `Reboot Devices` tab:

- click on the `+ Add a task` button (2) to add a new planification task then,
- click on the `+ Add a recurrence` button (3) to customize the planification recurrence.



To create a planification to set the screen of the devices registered in your project in standby, click on the `Switching device to standby mode` tab:

- click on the `+ Add a task` button (5) to add a new planification task then,
- click on the `+ Add a recurrence` button (6) to customize the planification recurrence,
- click on the `+ Add an exception` button (7) to add some Exceptions days when the devices must not be in standby.



Once the tasks are created or modified, the publication of the App on the devices is required so that the tasks are effective on the devices.

Part III

Project

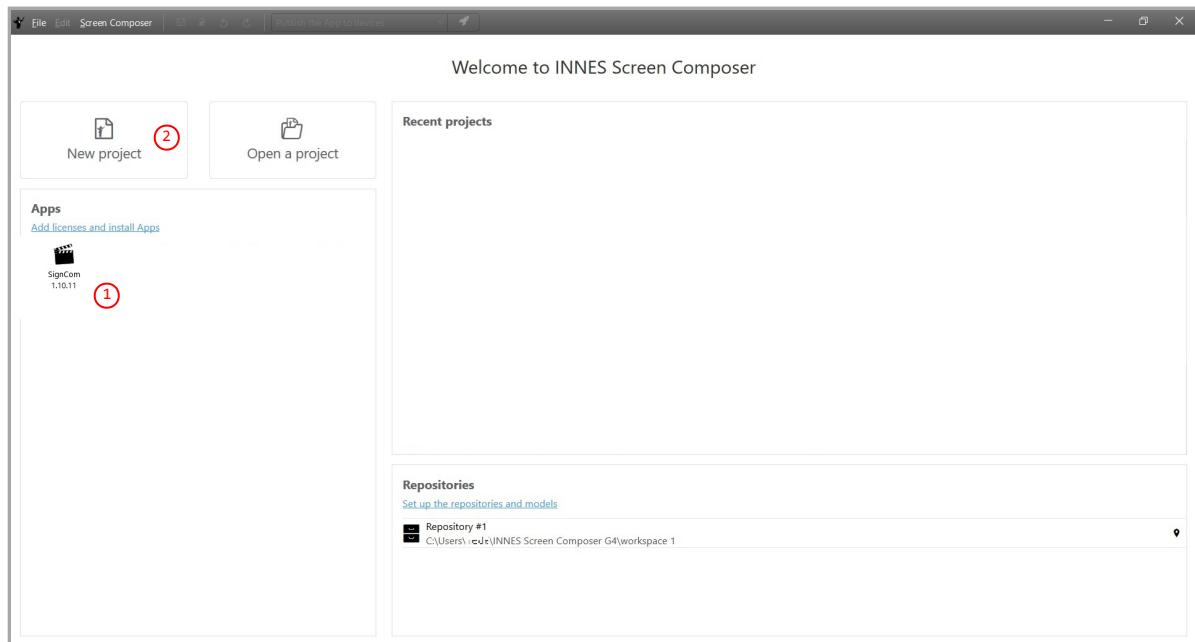
SignCom new project

Here is the SignCom App logo:



Once the SignCom App (1) is installed with its licenses keys:

- click on the New project (2) button (or the New project item from the File menu) or
- double click on the SignCom App (1) button.



Fill the free text label **③**.

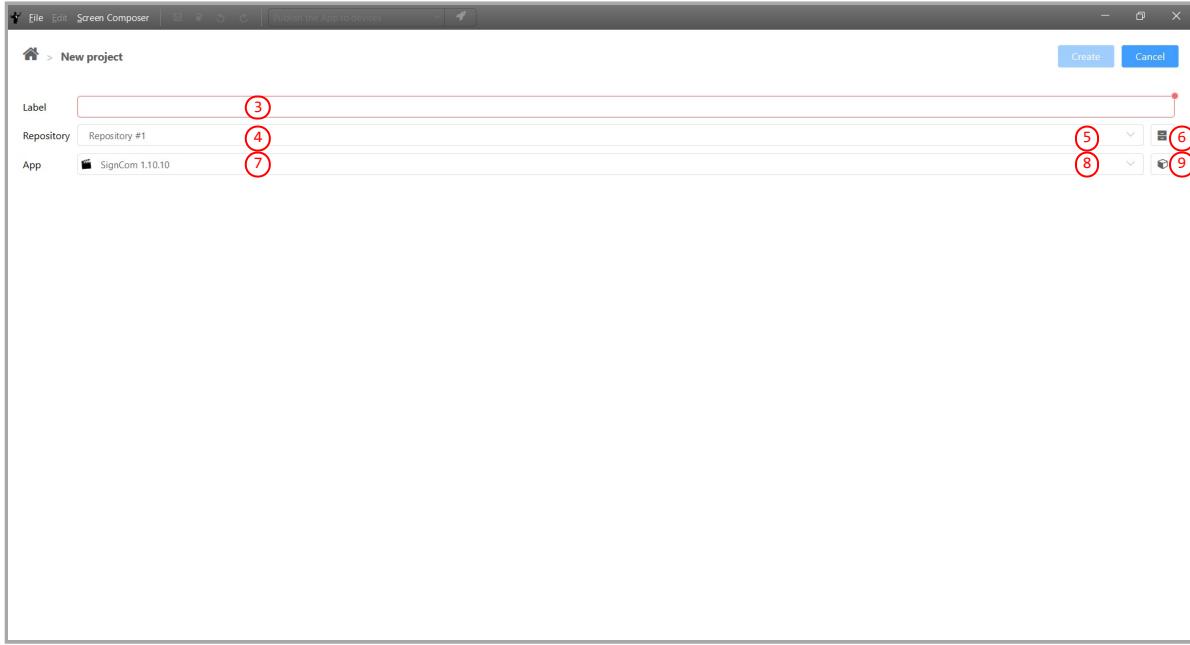
The default project repository is named **Repository #1** **④** and is located by default in **C:\Users\<username>\INNES Screen Composer G4\workspace 1**. You can decide to choose another one with the repository drop down list **⑤** or create another one by clicking on the **Edit repositories** **⑥** button. For further information, refer to the chapter § **Repositories**.

If only the SignCom App product is available, the SignCom **⑦** App product is automatically selected.

If you have several App products installed, use the App product's drop down list **⑧** to select the SignCom App product.

If it is not available, install the SignCom App product with the **⑨** button.

The last SignCom App version is available on the [INNES support Web site](#).



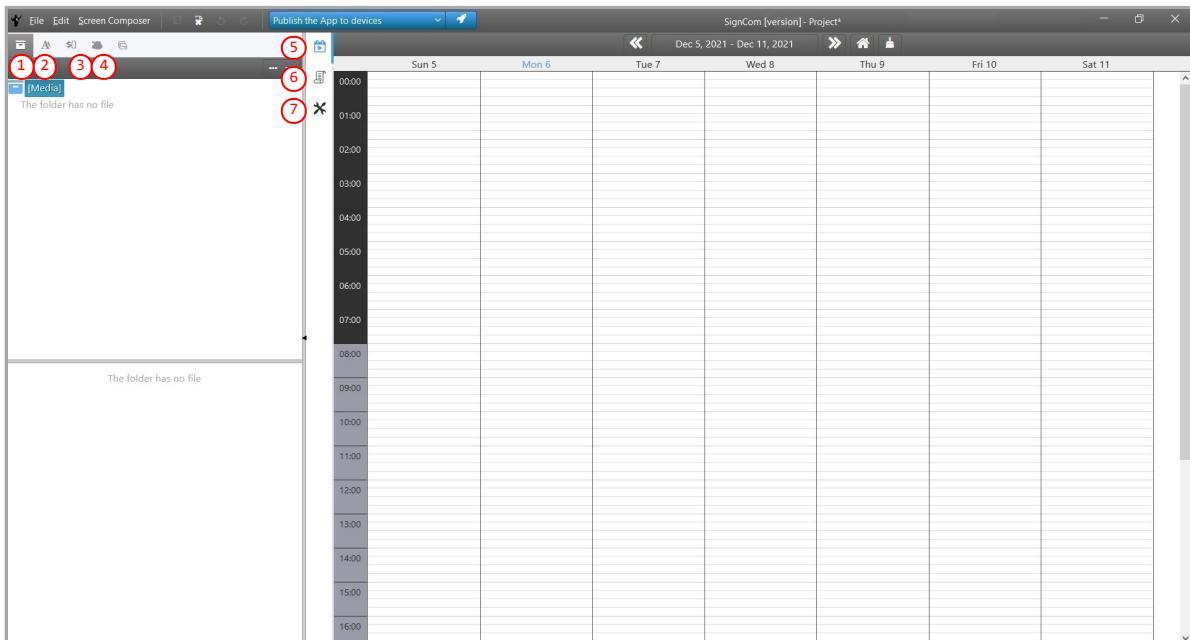
Click on the **Create** button.

The resources for your SignCom App project are located on the left side:

- **①** Medias
- **②** Fonts
- **③** Variables
- **④** Devices

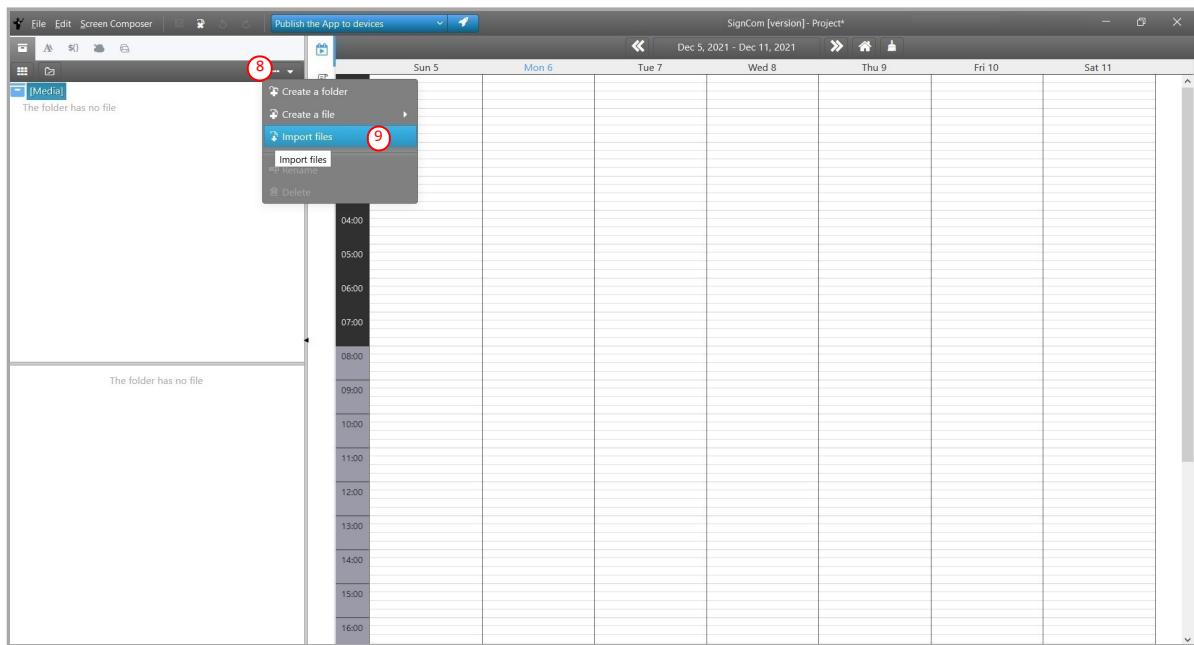
There are three main configuration tabs for your SignCom App project on the right side:

- **⑤** Visual channel calendar
- **⑥** Manifest
- **⑦** Properties



Once your project is created, you must add *devices* to your project and activate some of them as applicant for the publication. For further information, refer to the chapter § [File > Project settings > Devices](#).

To abond the medias library, you can import medias from your computer. Click on the three dots [...] [\(8\)](#) menu then click on the [Import files](#) [\(9\)](#) button.



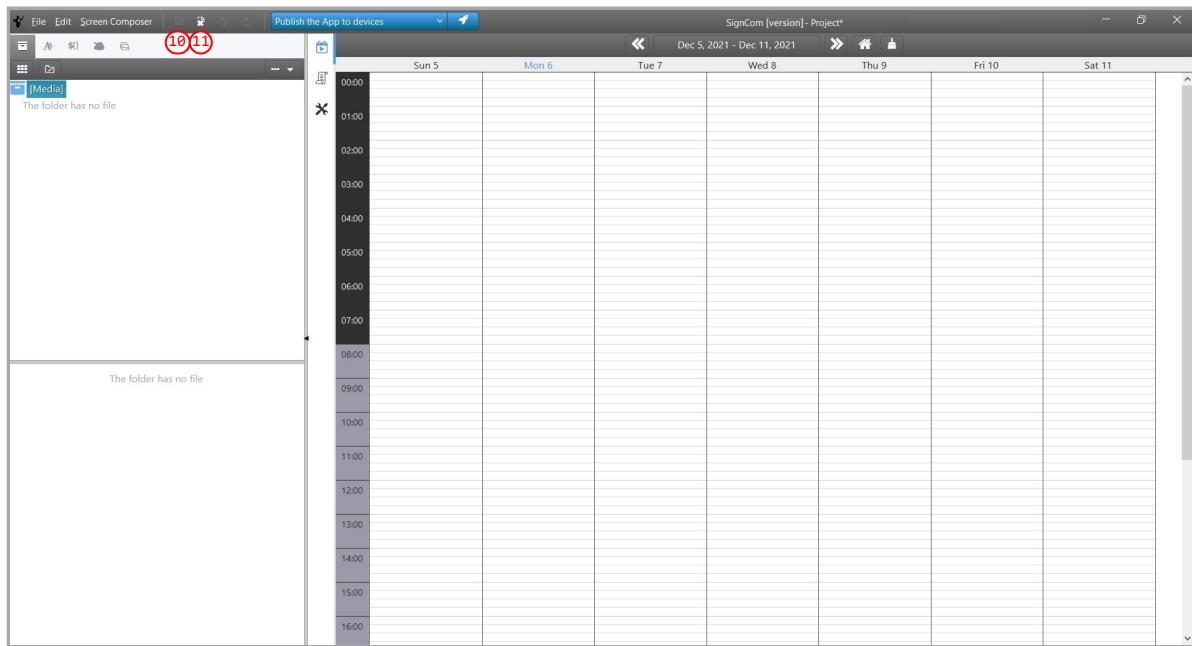
For further information, refer to the chapter § [Media](#).

SignCom project saving

To save your project into a `.isz` file:

- click on the `Save`  button,
- click on the `Save` item from the `File` menu,
- or `[CTRL] + [s]`.

 The project is not saved automatically. It is recommended to save it regularly.



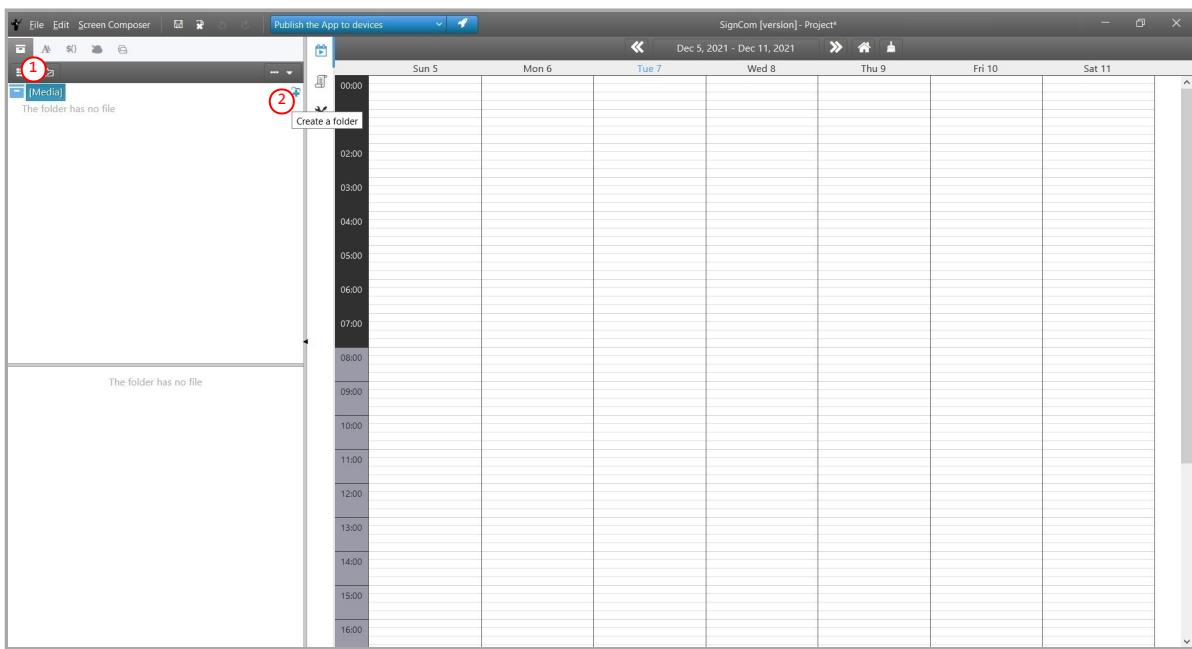
SignCom project closing

To close the project:

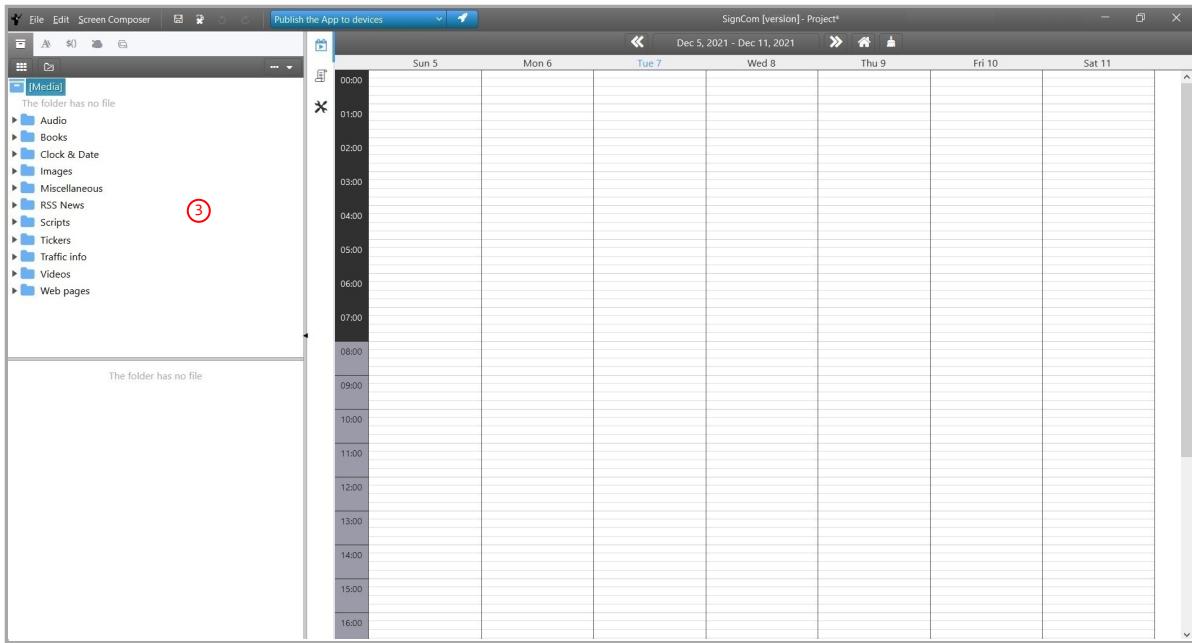
- click on the `Close`  button,
- or click on the `Close` button from the `File` menu.

3.2 Medias

Click on the **Media** (1) tab to view the **Media** files tree of your project repository. To organize your medias library, it is advised to create a folder tree. To create a folder, fly the mouse pointer over the button in front of **Media** root folder to make appear the **Create a folder** (2) button then click on it.

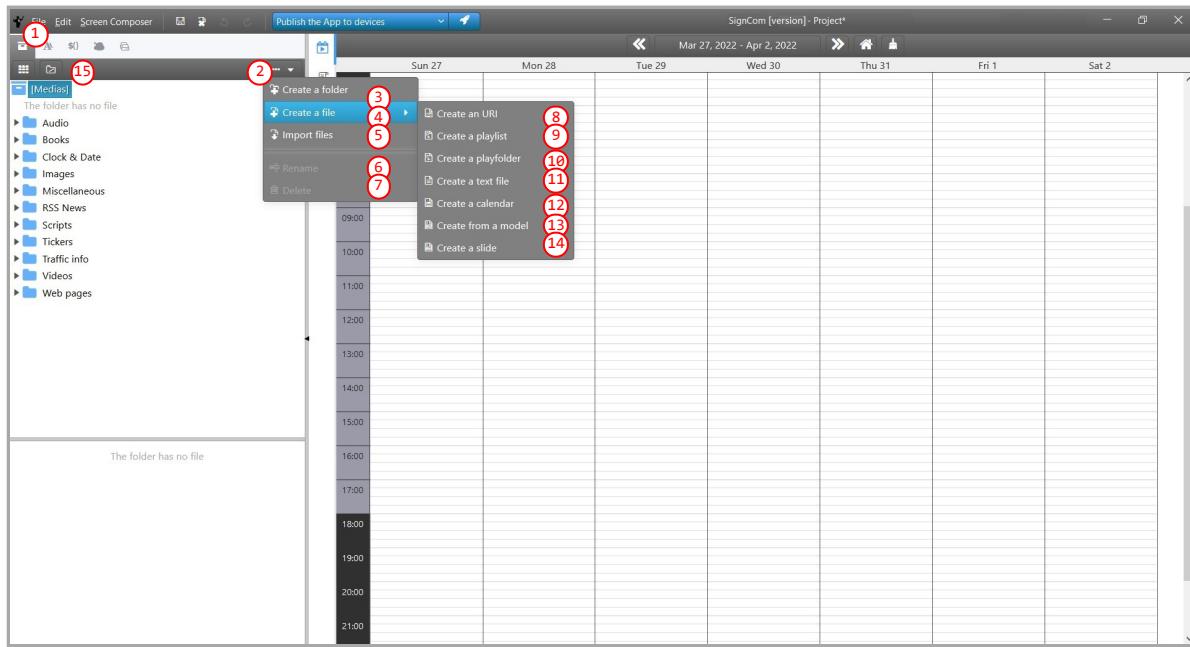


This is a folder organisation (3) example:



To abound the **Media** tab of your repository, select a folder then click on the three dots [...] **②** button which gives access to several items:

- **>Create a folder** **③**,
- **Create a file** **④**,
 - **Create an URI** **⑧**,
 - **Create a playlist** **⑨**,
 - **Create a playfolder** **⑩**,
 - **Create a text file** **⑪** (.css, .csv, .htm, .html, .ics, .js, .json, .md, .mdp, .sdp, .srt, .tsv, .txt, .vtt, .xml),
 - **Create a calendar** **⑫**,
 - **Create from a model** **⑬**,
 - **Create a slide** **⑭**
- **Import files** **⑤**: With the supported extensions .apng, .ASF, .avi, .css, .CSV, .EOF, .GIF, .HTM, .HTML, .ICS, .ISMC, .JPG, .JPEG, .JS, .JSON, .M2V, .M4A, .M4V, .MAF, .MAFF, .MD, .MJPG, .MKA, .MKS, .MKV, .MOV, .MP3, .MP4, .MPD, .MPG, .MPEG, .MXF, .ODT, .ODP, .OGA, .OGG, .OGV, .PDF, .PLAYFOLDER, .PLAYLIST, .PNG, .PPS, .PPSX, .PPT, .PPTX, .PS, .SDP, .SRT, .SVG, .TS, .TSV, .TTF, .TXT, .URI, .VOB, .VTT, .WEBM, .WEBP, .WGT, .WMA, .WMV, .WOFF, .XLS, .XLSX, .XML, .XSPF,
- **Rename** **⑥**,
- **Delete** **⑦**.



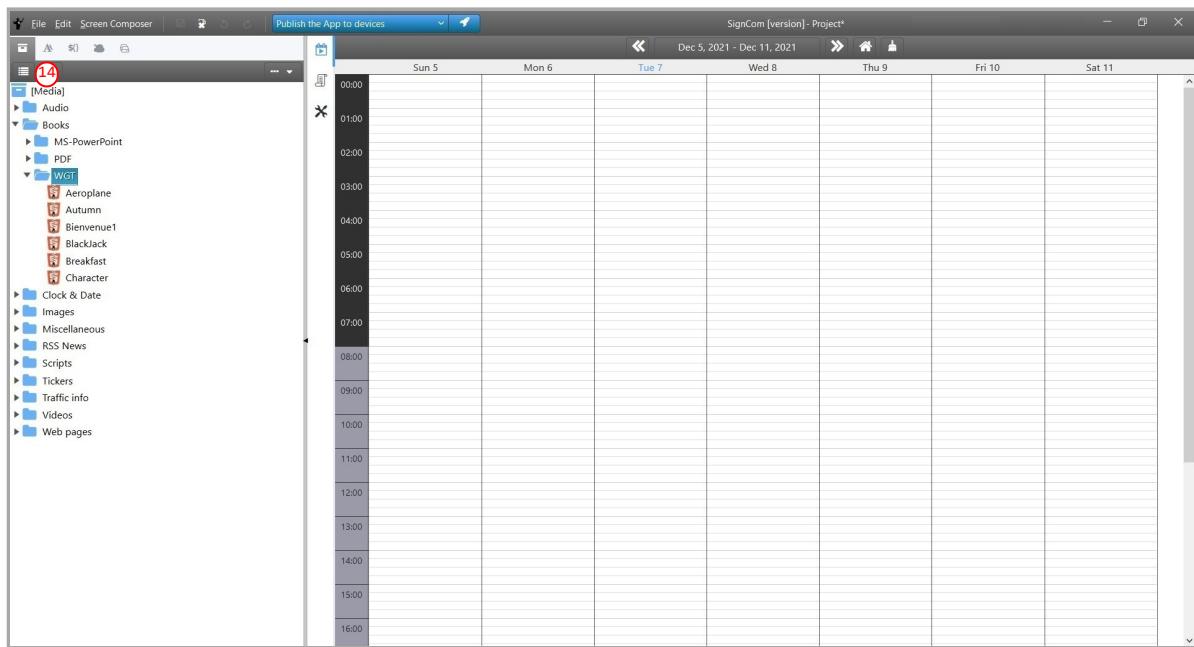
The **Collapse all folders** **⑯** **button is not supported.**

Organize the **Media** tab of your project repository by creating folders and abond it by simply importing medias:

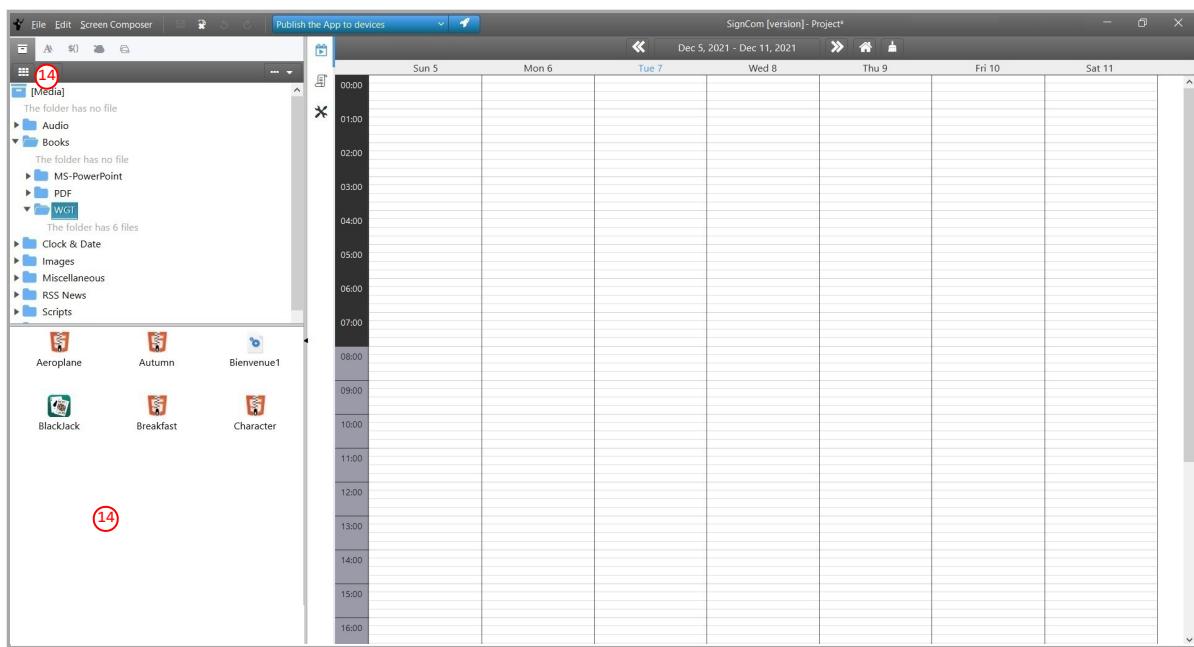
- videos media,
- images (background, company logo, channel logo),
- HTML widget (.wgt, .maff),
- MS-PowerPoint presentation (.ppsx, .pptx),
- PDF document (.pdf).

The *Display by thumbnail* / *Display by tree* toggle button allows to change the way to show the files tree.

Here is a files tree example when the *Display by tree* mode is selected.

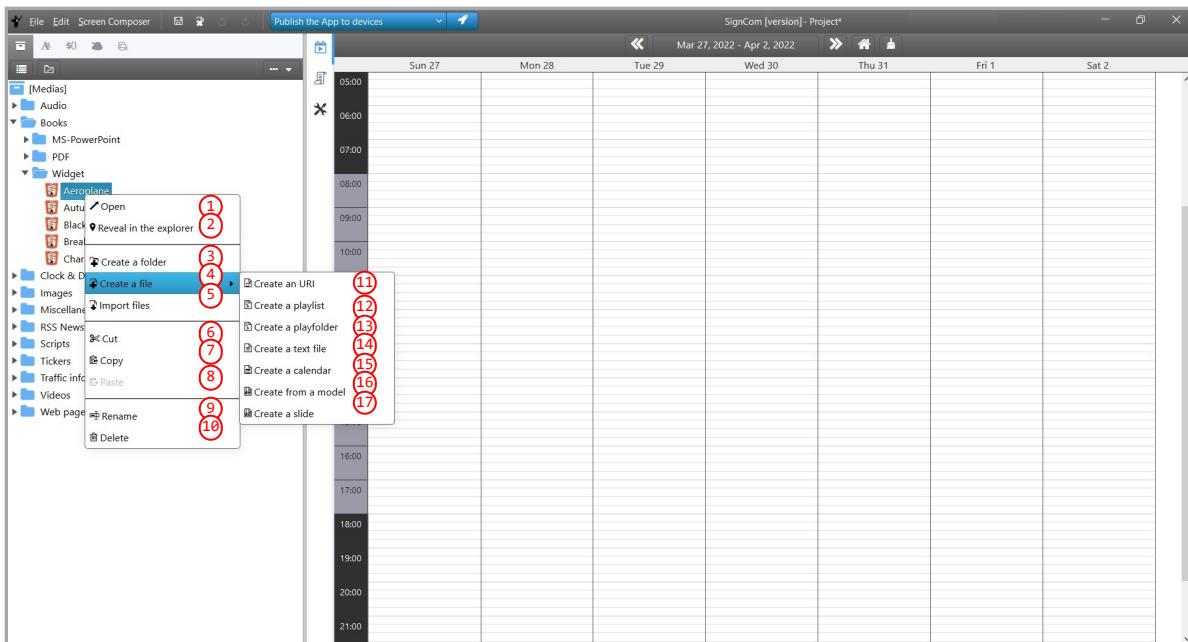


Here is a files tree example when the *Display by thumbnail* mode is selected.



To organize the files and folders in the **Media** tab of your project repository, select a file (or a folder), and make a right click to access to a contextual menu having the following items:

- Open (1): allows to open the file on the right side to preview it, add metadata or edit it (for file only),
- Reveal in the explorer (2): allows to see the file (or the folder) location in a MS-Explorer windows and set the focus on the file (or the folder),
- Create a folder (3),
- Create a file (4):
 - Create an URI (11): allows to create Web page URI,
 - Create a playlist (12): allows to create medias playlists based on rules like media rating criteria,
 - Create a playfolder (13): allows to create medias playfolder based on rules like media rating criteria,
 - Create a text file (14),
 - Create a calendar (15),
 - Create from a model (16): allows to create HTML widget based on content models and URIs based on URI models,
 - Create a slide (17): allows to create a slideshow based on a programmable timeline,
- Import files (5),
- Cut (6),
- Copy (7),
- Paste (8),
- Rename (9),
- Delete (10).



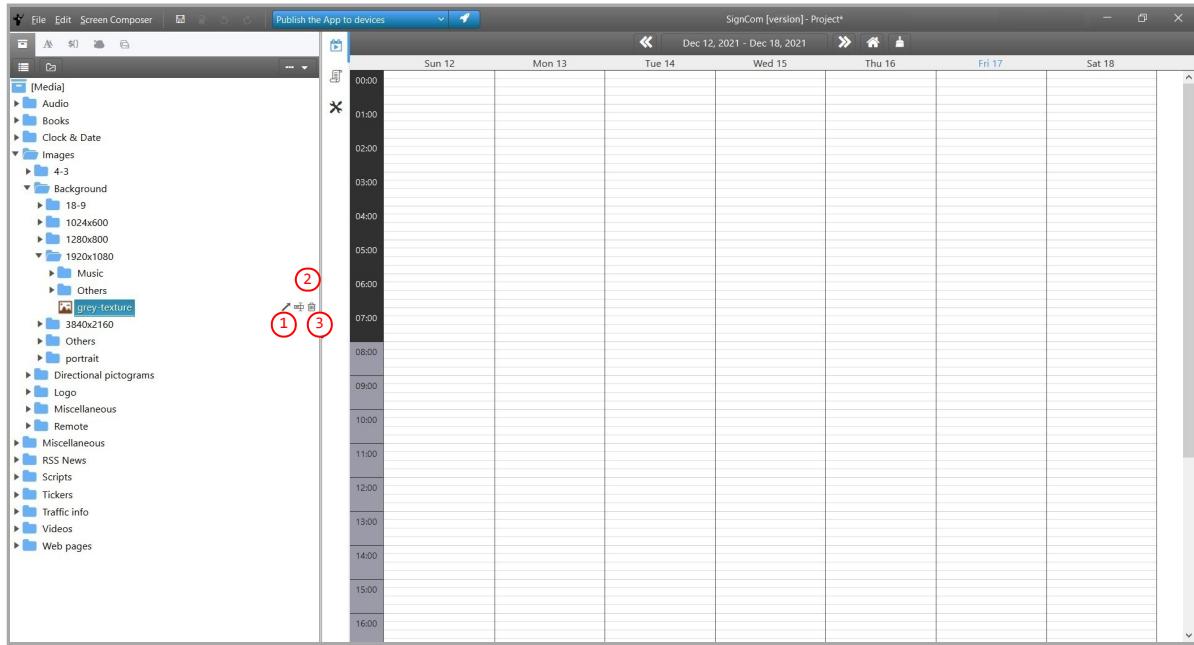
3.2.1 Medias > Preview and metadata

Opening a media from the library allows to display a window on the right allowing to:

- view/modify/add the metadata of the file,
- launch a unitary preview of the media.

Select a media in the **Media** tab, then fly the mouse pointer over it, on its right, to make appear three contextual buttons, which are shortcuts to some items of the file contextual menu:

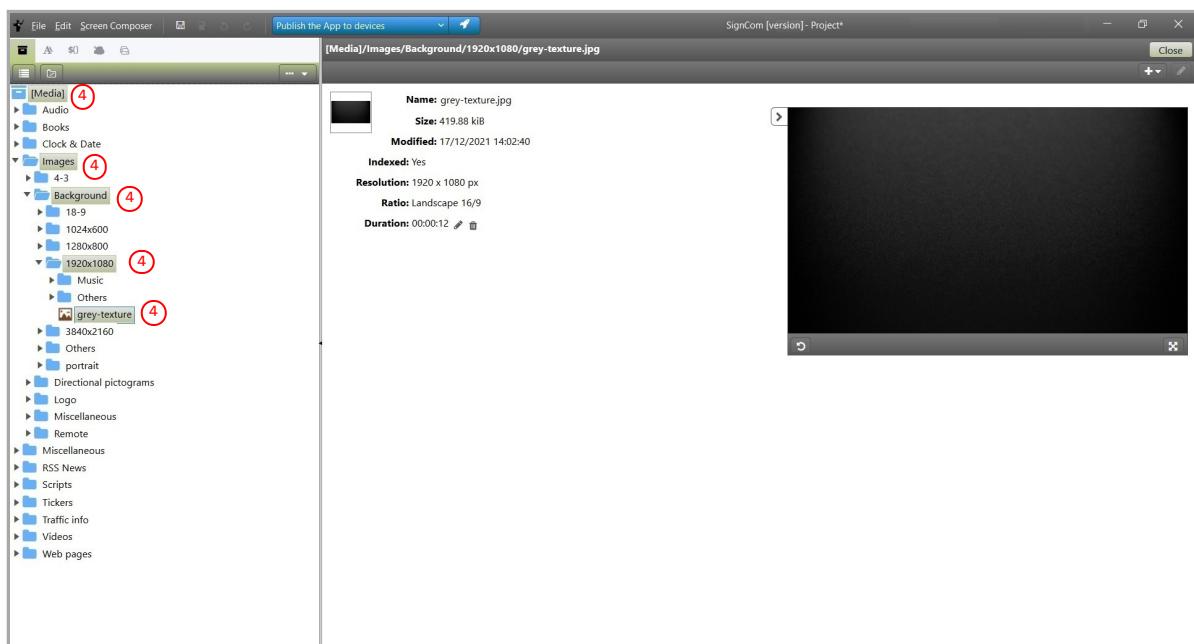
- Open (1),
- Rename (2),
- Delete (3)



Click on the Open (1) button to open the unitary preview of the media on the right pane.

When a file is already opened but not being edited, clicking on the Open (1) button of another file, closes automatically the previous one and opens the new one.

It is not possible to move a file to another folder if this file is currently opened. It is not possible to move a folder to another one if it contains a file which is currently opened. The filename of an opened media cannot be renamed. To inform the user that a file is being edited, the filename and all the folder of the path filename are highlighted (4) in a green color style.



To close the file to return to the **SignCom** App view on the right side, click on the **close** button.

Default metadata

When the media is opened, its metadata can be watched at first glance. The number of displayed metadata depends on the opened media type.

The following metadata can be found for all files:

- Name (1): name of the file with its extension,
- Size (2): memory footprint of the media on your storage space,
- Modified (3): date of the last file modification,
- Indexed (4): indicates whether the media has been successfully indexed in the database.

The Thumbnail (5) metadata can be edited for all previewable medias except:

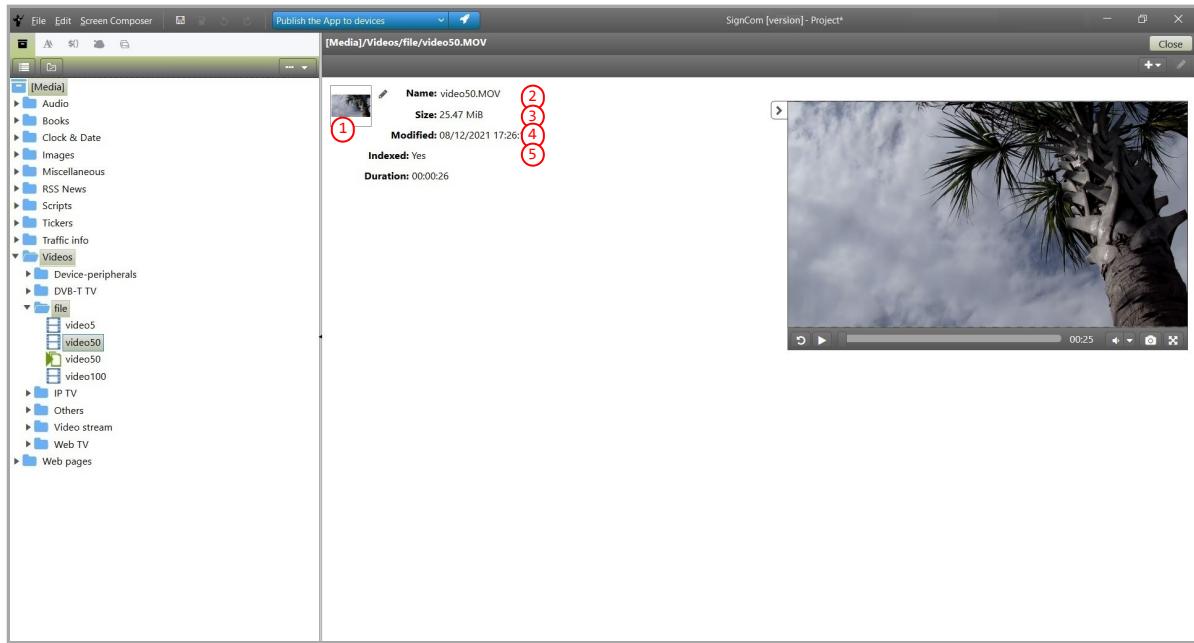
- images and
- MS-PowerPoint presentations.

[The supported formats for the thumbnails are: .png, .svg, .gif, .jpg, .jpeg.]

[Once loaded in the medias, the resolution of the thumbnail is downscaled to a height of 120 px max. and a width of 67 px max.]

To modify the thumbnail, click on the Edit (1) and choose your own picture.

Example with a video:



The number of available metadata depends on the type of the media.

Unitary preview

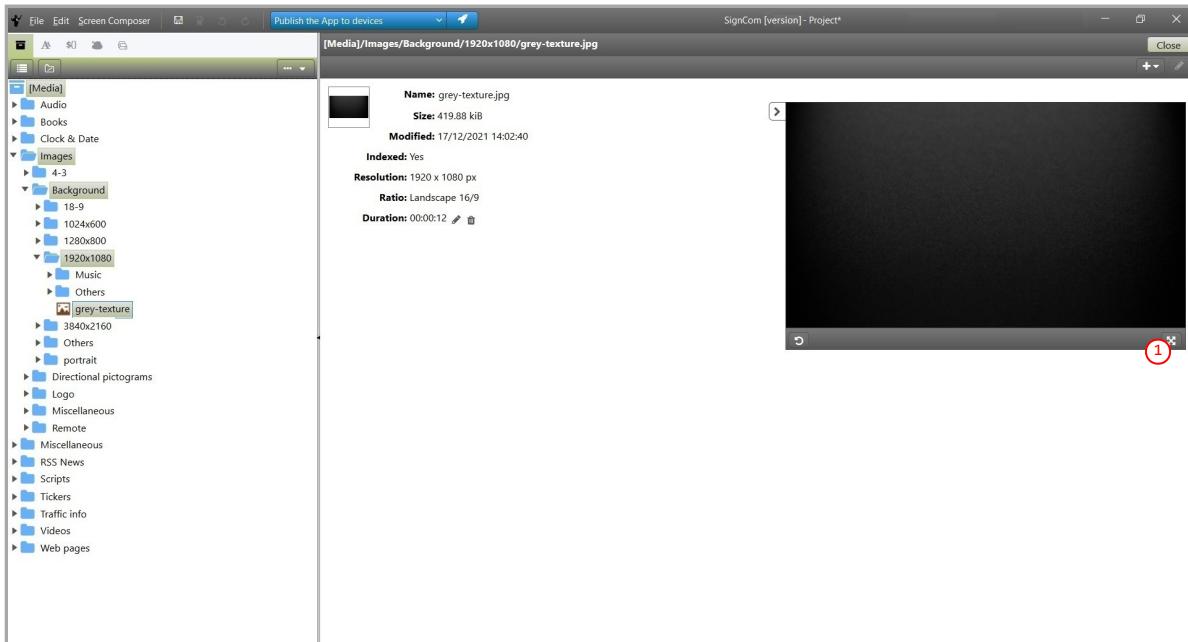
The unitary preview of images is supported for these formats:

- .png ,
- .apng ,
- .svg ,
- .gif ,
- .jpg ,
- .jpeg .

 The support for unitary preview of .webp image will be supported in a next version.

The metadata specific to the images media are:

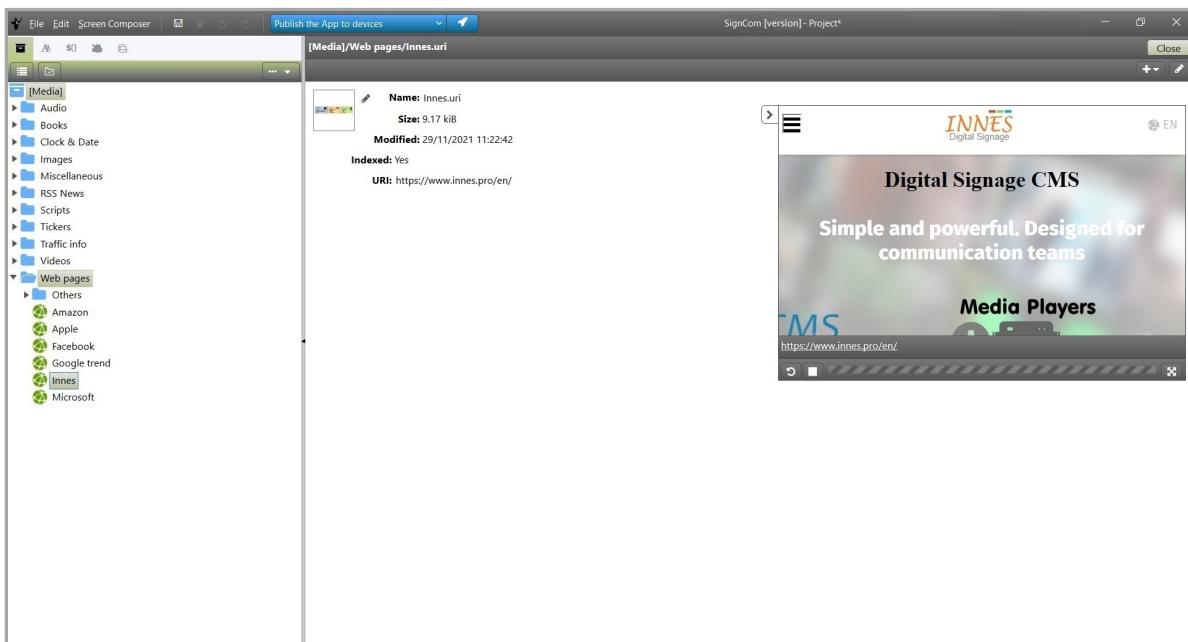
- **Resolution**: width and height of the media in pixels,
- **Ratio**: the values *Landscape 16/9*, *Portrait 9/16*, *Landscape 4/3*, *Portrait 3/4* appear when the resolution is strictly 16/9, 9/16, 4/3 or 3/4.



Click the **Expand** button  (1) to preview the media in a full screen preview window.

Click on the  button to launch the rendering of your favorite Web pages URI or Web TV URI with the `http://` or `https://` scheme.

For further information about URI creation and edition, refer to the chapter § [Medias > URI](#).



Click on the ► button to launch the unitary preview of your video having these container extensions:

- .m4v ,
- .mp4
- .mov .

 The support for unitary preview of .webm container will be supported in a next version.

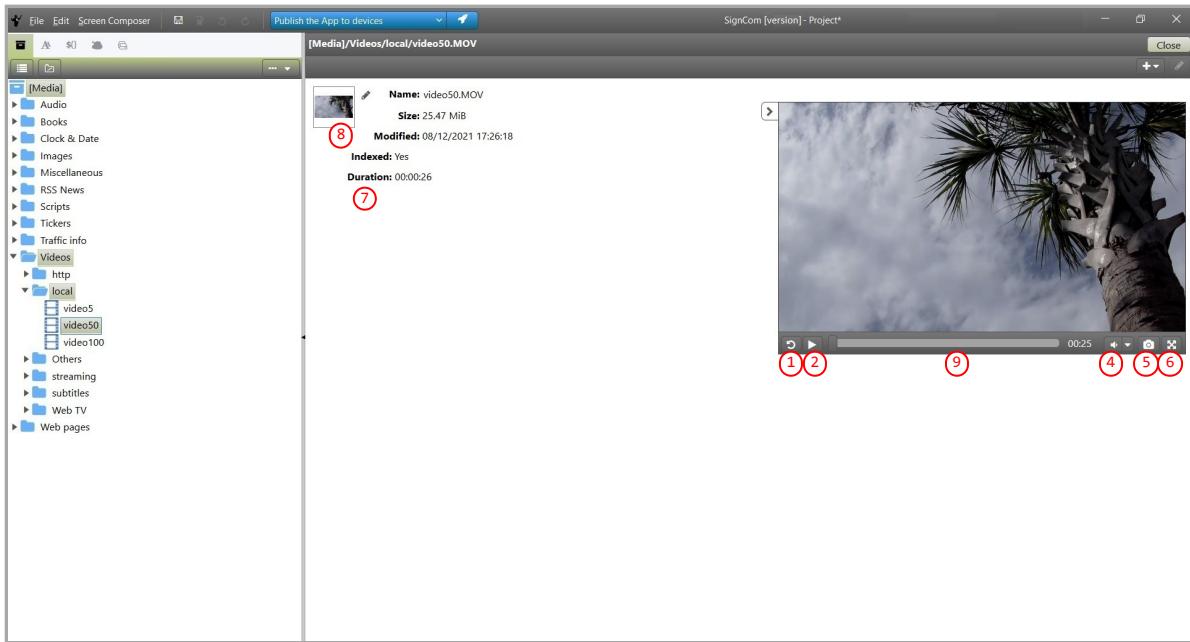
The availability of some buttons on the preview toolbar depends on the type of media to visualize.

For a video, these are the available buttons:

-  Restart (1) to move to the beginning of the media,
-  Play (2) to launch the media unitary preview,
- time scrollbar (3): drag the scrollbar to reach a particular video image.
-  Volume (4) to increase or decrease the volume while the video is played,
-  Choose this picture as thumbnail button (5) to choose the current display frame of the video, at a defined index of the video timeline, as media thumbnail (8),
-  Expand (6): to preview the media in full screen,
- The bargraph (6) allows to start the video at a particular timestamp.

When the preview window is displayed in full screen, the Minimize button (7) reduces the size of the preview window.

The metadata specific to the video (.m4v, .mp4 and .mov) is the Duration (7).



The playback of audio files is also supported for these file extensions:

- .m4a ,
- .mp3 .

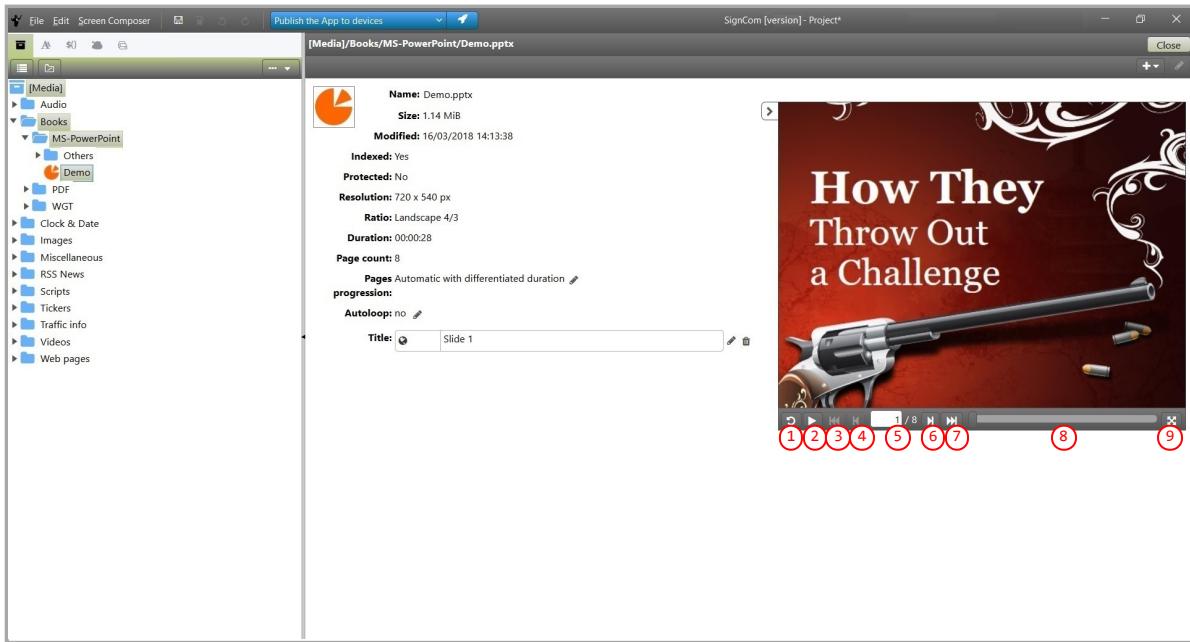
Click on the ► button to launch the unitary preview of your MS-PowerPoint presentations and your PDF documents having these file extensions:

- .pptx ,
- .ppsx ,
- .pdf .

In case you have .ppt or .pps MS-PowerPoint presentations, it is recommended to save them into .pptx or .ppsx with your MS-Office software suite so that they are supported in this unitary preview window, in the calendar time slot preview window and by Gekkota 4 devices.

Click on these buttons to browse the different pages of the document:

- ⏪ Restart (1): to start the media preview at the beginning of the media,
- ► Play (2) button to launch the media unitary preview window,
- ⏴ Start (3): preview only the first page,
- ⏵ Previous (4): to preview only the previous page,
- page number input (5): allow to preview a specific page number
- ⏵ Next (6): to preview only the next page,
- ⏵ End (7): to preview only the last page.
- The scrollbar (8) allows to drag to a specific page.
- ⏷ Expand (9) to preview the media in full screen,



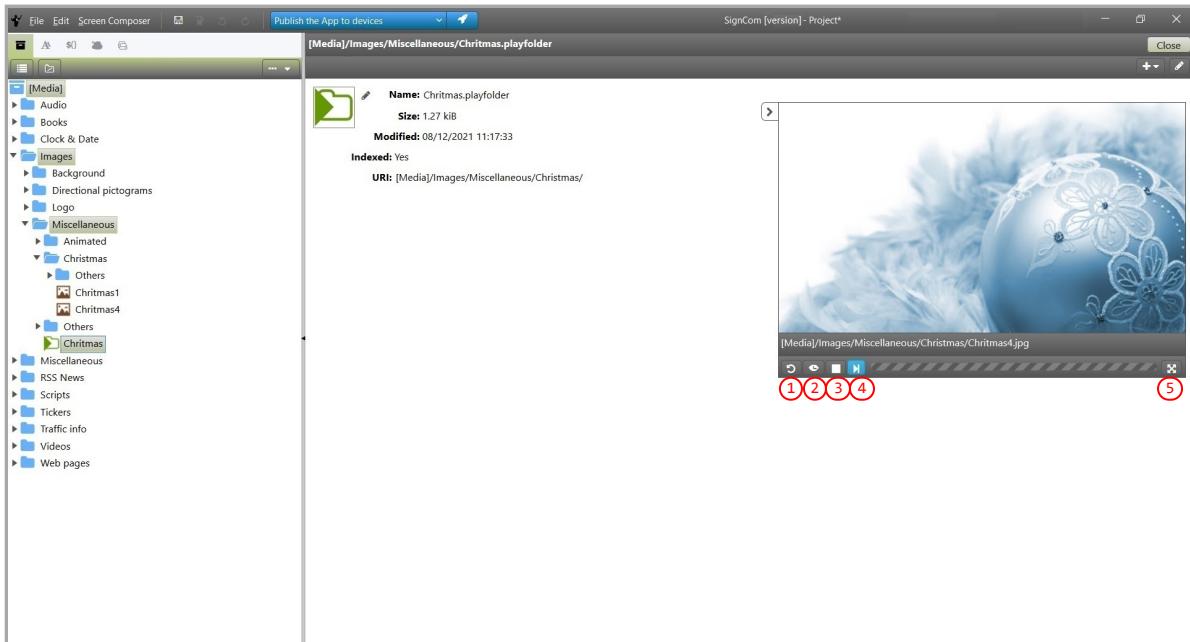
Click on the ► button to launch the unitary preview of your playfolders or playlists having these extensions:

- .playfolder ,
- .playlist ,
- .xspf .

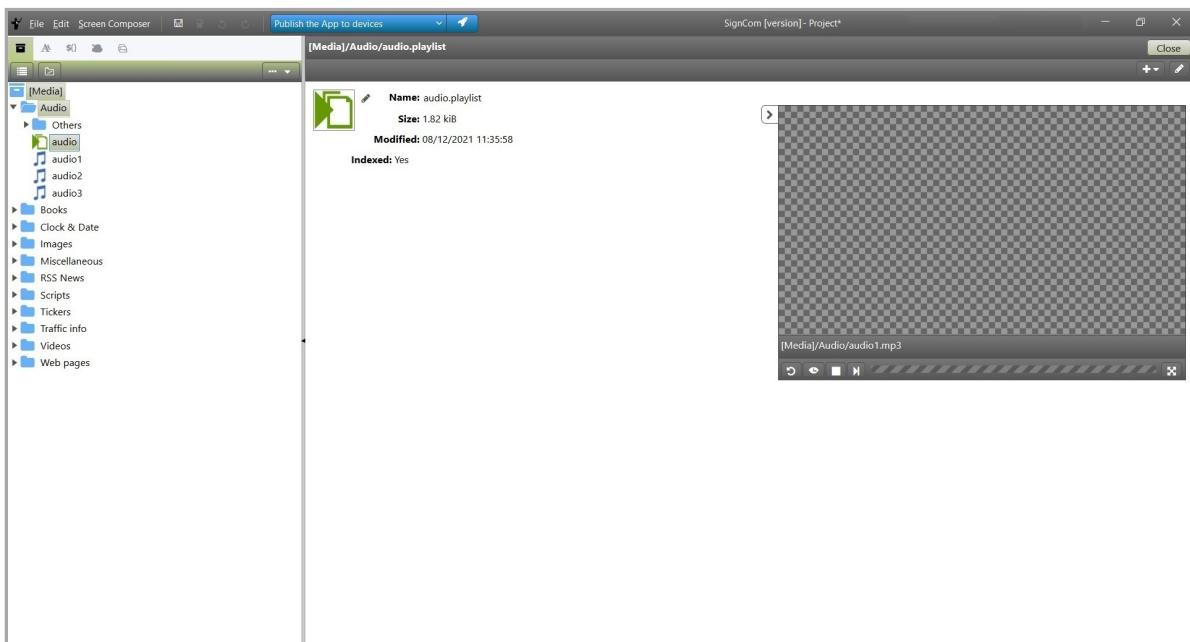
☞ When a playlist is created with INNES Screen Composer 64 , it can be created with the .playlist extension only.

Click on these buttons to browse the different medias played by your playlist or your playfolder:

- ⏪ Restart (1): restart the unitary preview window to the beginning,
- ⏴ Date and time of preview dd/mm/yyyy at hh:mm (2): allows to select the start time for the preview window, useful when the playfolder or the playlist play medias having a validity dates,
- ⏹ Stop (3): stop the unitary preview,
- ⏷ Next (4): preview the next page,
- ⏷ Expand (5) button to launch the preview window in full screen.
- The bargraph allows to start the video at a particular timestamp.



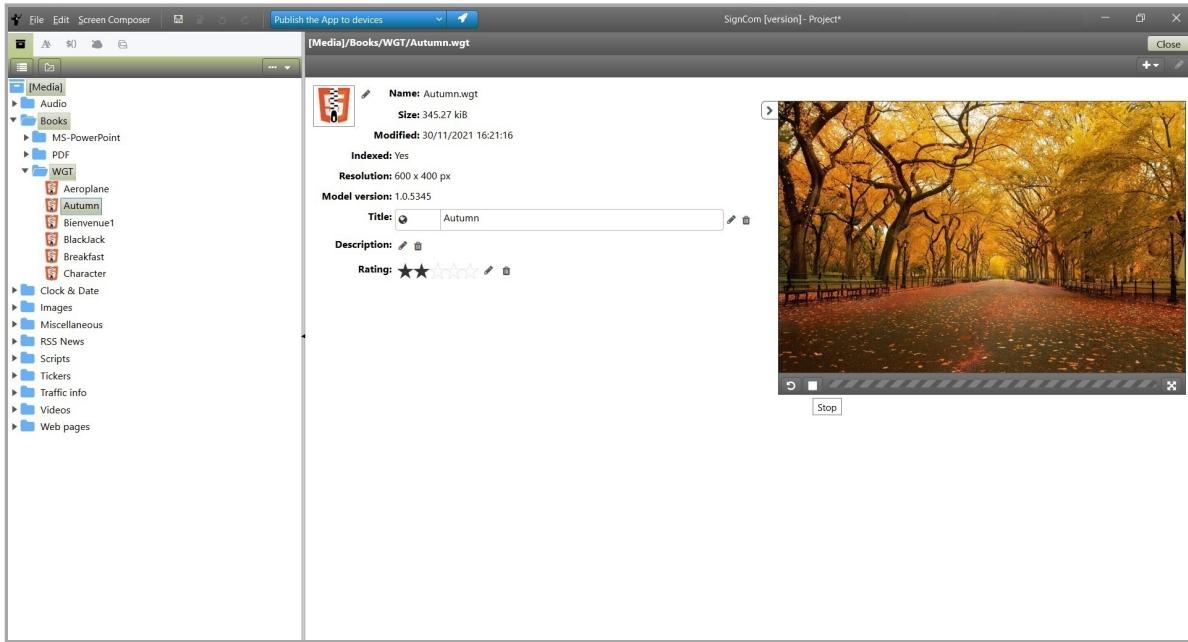
For further information about playfolder creation and edition, refer to the chapter the chapter § [Playfolder creation and edition](#).



For further information about playlist creation and edition, refer to the chapter the chapter § [Playlist creation and edition](#).

Click on the ► button to launch the unitary preview of HTML widgets having these file extensions:

- .maff ,
- .wgt .



Depending on the previewed media, the duration of the entire media preview can be more and less long. For some of them like URIs, it never ends. That is the same once published on the device. This table explains the good practise on how to ensure that a media is ending after a while.

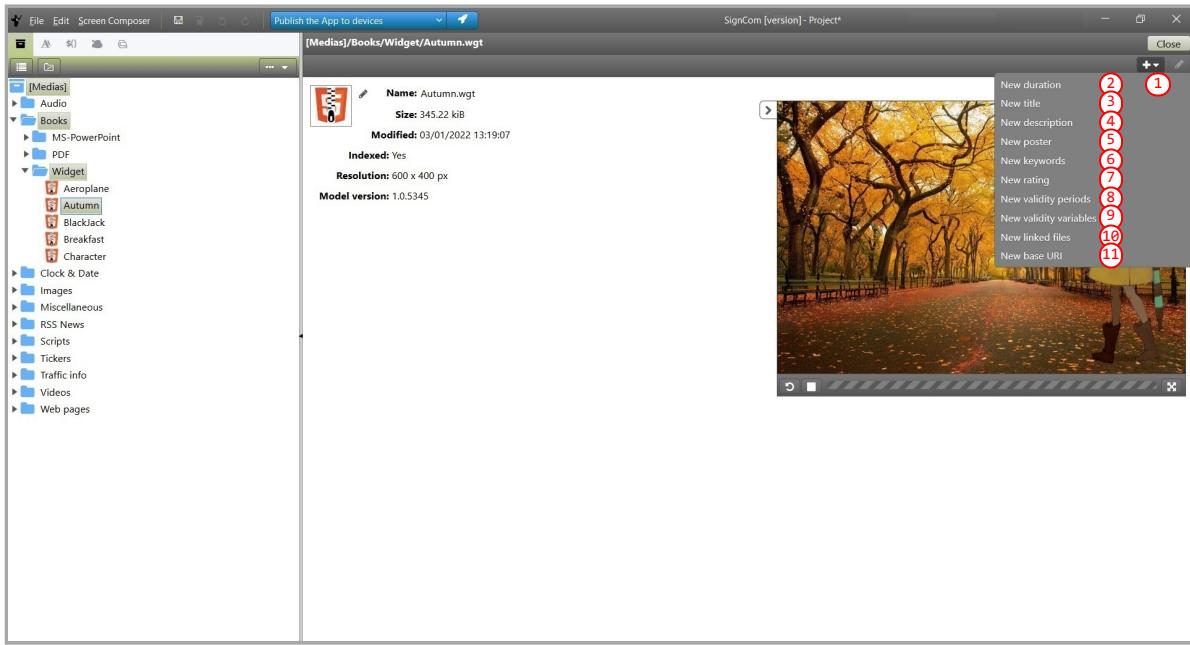
In case a media has an *intrinsic duration*, that means that it is ending naturally after a while.

Medias type	Intrinsic duration	Medias ending management
- video, - audio	yes	
- images	no	- by adding a value in second for the duration metadata.
- URI	no	- by adding a value in second for the duration metadata.
- playlist - playfolder	no	- is depending on the amount of medias to play and on the current date & time in case validity date metadata is used or, - by adding a value in second for the duration metadata.
- PDF documents, MS-PowerPoint presentations	yes (when the page progression for the page is never manual) (= duration per page X number of pages)	when the page progression for some page is manual - activate the Enforce on media with pagination a page progression with a default page duration SignCom option.

For further information about media ending management for HTML widget and URI based on model, refer to the chapter § [Appendix: Medias ending management for HTML widgets and URI based on models](#).

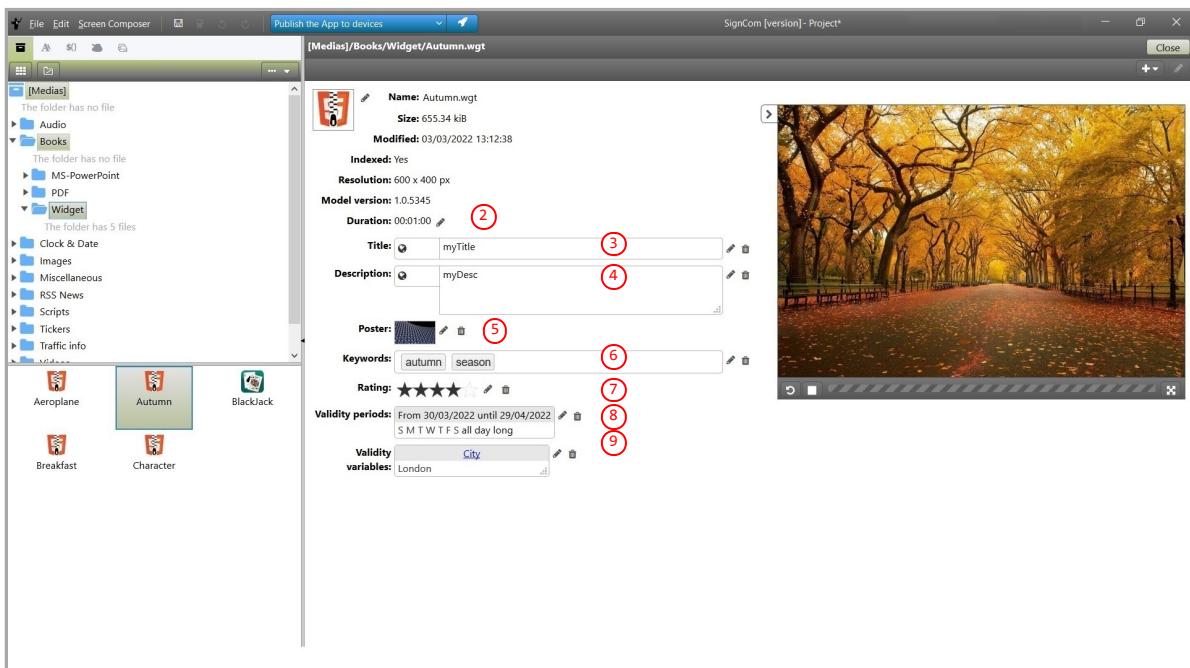
Adding additional metadatas

When they are not already present, it is possible to add additional metadatas to your media with the **+ Create a metadata property** (1) button.



- New duration (2): allows the media to quit after a defined duration,
- New title (3): allows to add standardized titles , language level dependant:
 - the first level is corresponding to the international language (most often English),
 - the following ones are corresponding to the desired languages,
- New description (4) : allows to add standardized descriptions , language level dependant:
 - the first level is corresponding to the international language (most often English),
 - the following ones are corresponding to the desired languages,
- New poster (5): allow to add a image to the media³
- New keywords (6): allows to associate keywords to your media. These keywords can then be used in the playing rules in playfolders or in playlists.
- New rating (7): allows to associate a rating to your media from 1 to 5 stars. This rating can then be used in the playing rules in playfolders or in playlists,
- New validity periods (8) : allows to associate a validity period to the media that is used during to evaluate whether the media can be played of not,
- New validity variables ² (9): allows to associate a validity variables to the media (such as Calendar , Text , or File Name). It is useful for example to apply a different playing rule depending on the location of a range of devices.
- New linked files (10): allows to add dependencies resources file to the medias if required by it.
- New base URI (11): allows to change the base of dependencies resources files when the media is using resources file with relative path.

This is an example of media when several metadatas have been added.



¹ Max. 260 characters per keyword. The capital letters of keywords are converted to lower case. During a CTRL+V, the keyword is automatically validated. When the ";" key is entered, the keyword is automatically validated. To set several keywords at a time for media, edit the keyword metadata and paste the keywords values separated by ; (e.g. keyword1;keyword2;keyword3;keyword4).

² If several validity variables values are added to the media in the validity variables metadata, if only one of the values is present on the device, the media is played.

³ The supported formats for the media's Poster metadata are: .png , .apng , .gif , .jpg , .jpeg . When loaded into the media's Poster metadata, the image is reduced to the same aspect ratio with a height of 1920 px max and a width of 1920 px max. The Poster metadata is used in App other than SignCom.

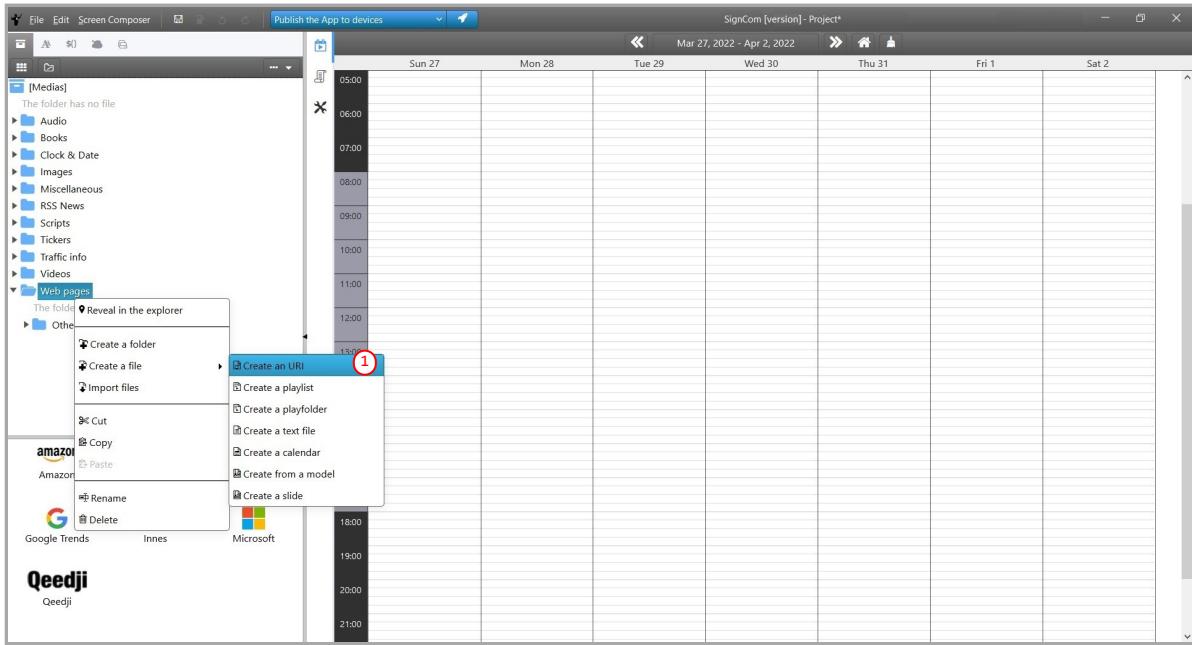
3.2.2 Medias > URI

The `URI (.uri)` is the way to create shortcuts to some Web content, to some remote content or to some local content like:

- Web page `URI`¹,
- DVB-T `URI`,
- Web TV `URI`,
- device peripherals `URI` for HDMI-input (`urn:`),
- IP TV `URI`, remote media `URI` (`http://`, `https://`),
- video stream `URI` (`rtp://`, `rtsp://`, `udp://`),
- `URI` shortcut to local media of your library (`[Medias]/`).

¹ Some credential types are supported in `URI`. That requires to create first a simple Web server services account.

To create an `URI`, select an appropriate folder and right click on it to display its contextual menu. Select the  `Create a file` item then click on the  `Create an URI` item. Enter a label for the `URI` and click on the `ok` button.



To fill the `location` input of the `URI` form, you can:

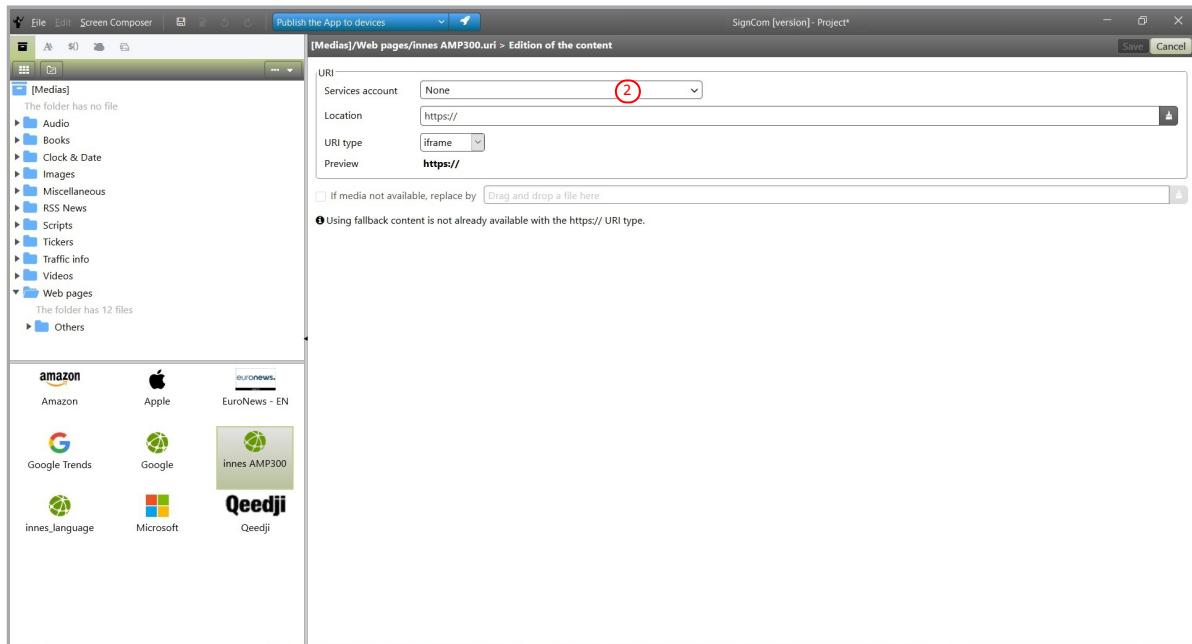
- either enter your favorite URL manually,
- or paste your favorite URL with its scheme.

 At the end of the operation, click on the `close` button to close the file and bring up the App SignCom configuration form.

Enter manually the URL

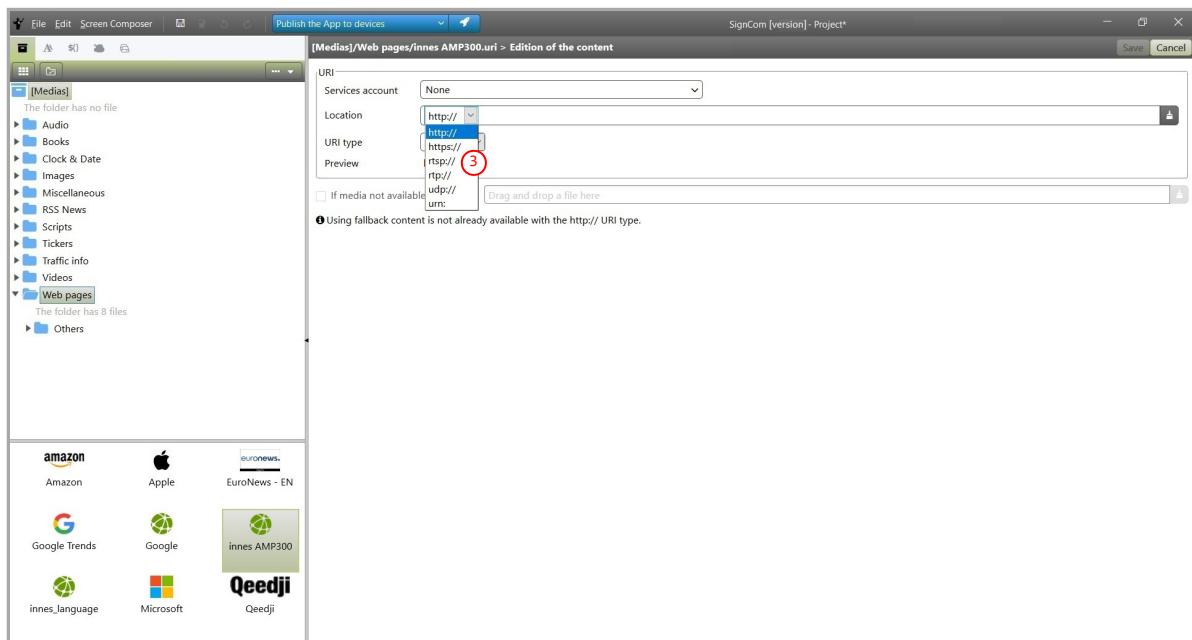
Edit your new URI.

The default services account value is None (2)

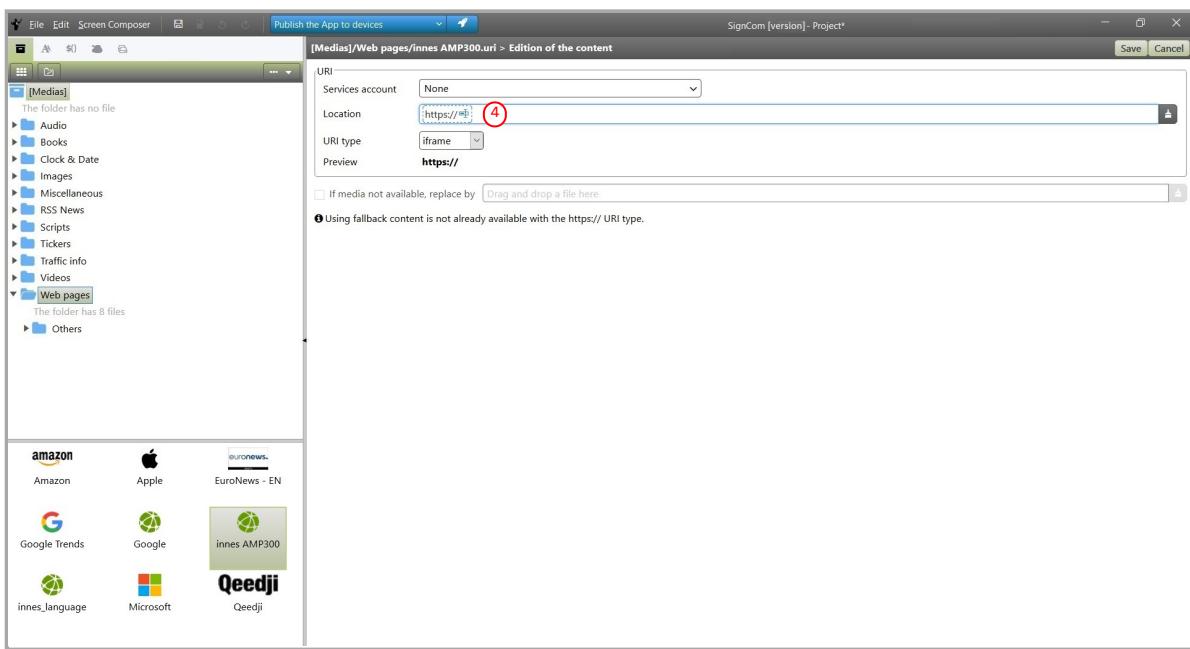


In the location input, click on the https:// default scheme to select another scheme among the following values (3):

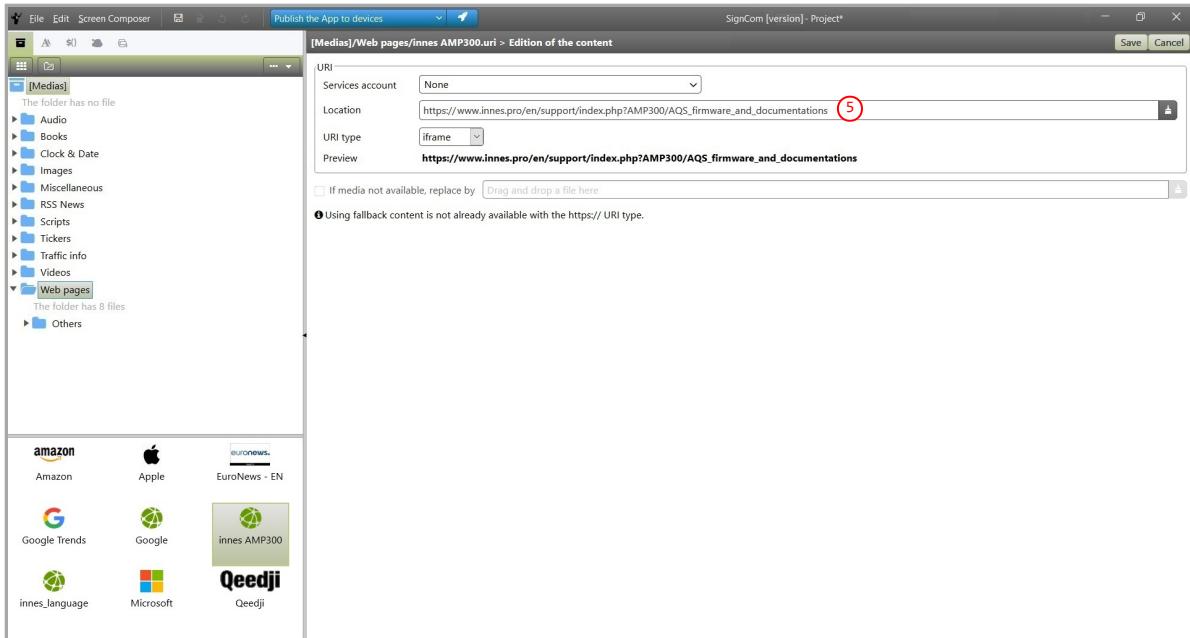
- http:// ,
- https:// ,
- rtsp:// ,
- rtp:// ,
- udp:// ,
- urn: .



Click in the `location` input. Fly the mouse pointer over the scheme to make appear the  blue button then click on it.



Enter the rest of your URL  then click on the `Save` button.

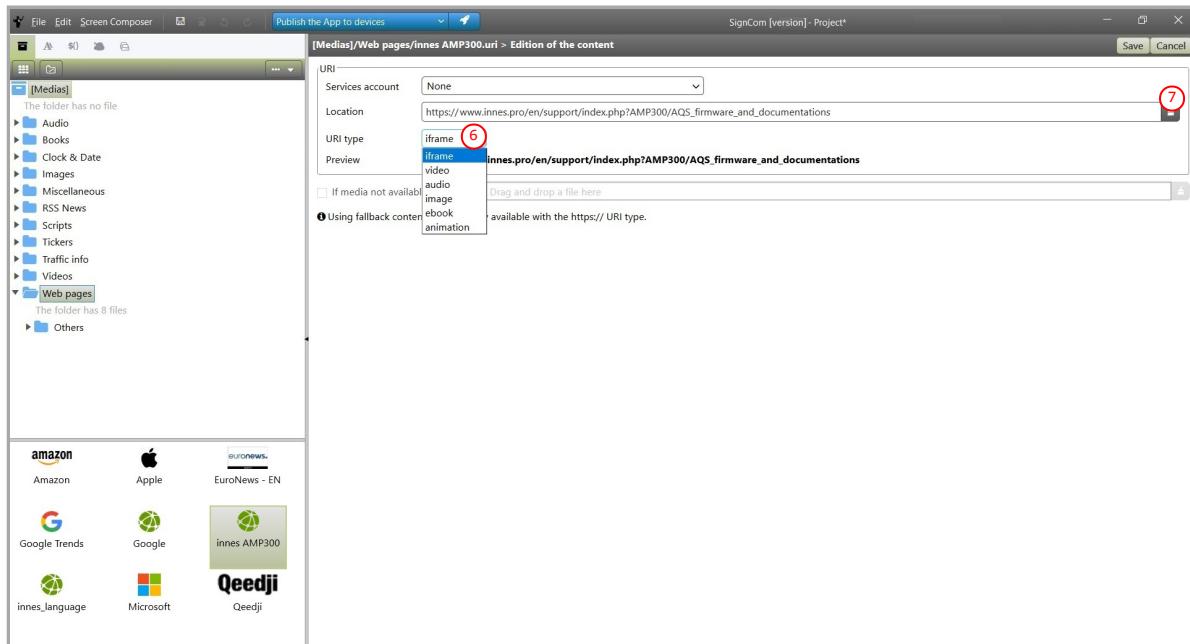


URI type

Select the appropriate URI type (6) among the following values:

- `iframe` (Web page),
- `video` (video stream),
- `audio` (audio stream),
- `image` (remote image),
- `ebook` (remote ebook),
- `animation` (remote flash animation¹).

¹ The Flash animation is deprecated.



Defining a default media that is played in case the URL is not available is supported only when the [Medias]/ scheme is used.

Clear URL input

To clear the location input, click on the (7) button on the right.

Web page URI, IP TV URI

These are URI examples showing the URI type used:

URI media	scheme generally used	URI type	URL example
- Web page URI - Remote PDF - Remote MS-PowerPoint - Remote images	<code>http://</code> <code>https://</code>	<code>iframe</code>	<code>http://224.2.15.10:1234/dir/doc.pdf</code> <code>https://www.innes.pro</code>
- Audio stream	<code>http://</code> <code>https://</code>	<code>audio</code>	<code>udp://239.2.15.10:1234</code>
- Video stream - IP TV URI	<code>http://</code> <code>https://</code> <code>rtsp://</code>	<code>video</code>	<code>udp://224.2.15.10:1234</code> <code>rtsp://192.168.1.16:1935/vod/video.mp4</code> <code>http://192.168.1.17/video.sdp</code> <code>rtsp://user:pass@192.168.60.108:554/realmon?channel=1&subtype=0</code>

The smb:// scheme is not supported in URI.

DVB-T URI

To create a DVB-T URI :

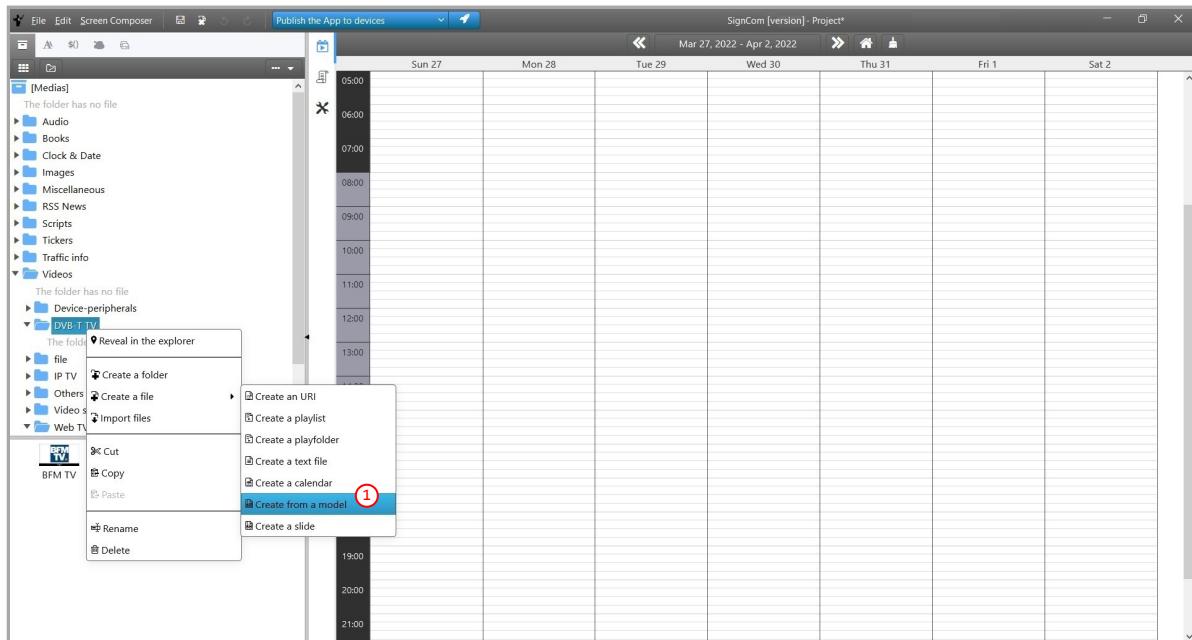
- create an URI from a DVB-T URI model¹,
- create an URI by entering the DVB-T URL manually with its urn: protocol,
- create an URI by pasting the DVB-T URL (urn: protocol included).

Note DVB-T URI are suitable when your device has a DVB-T tuner card.

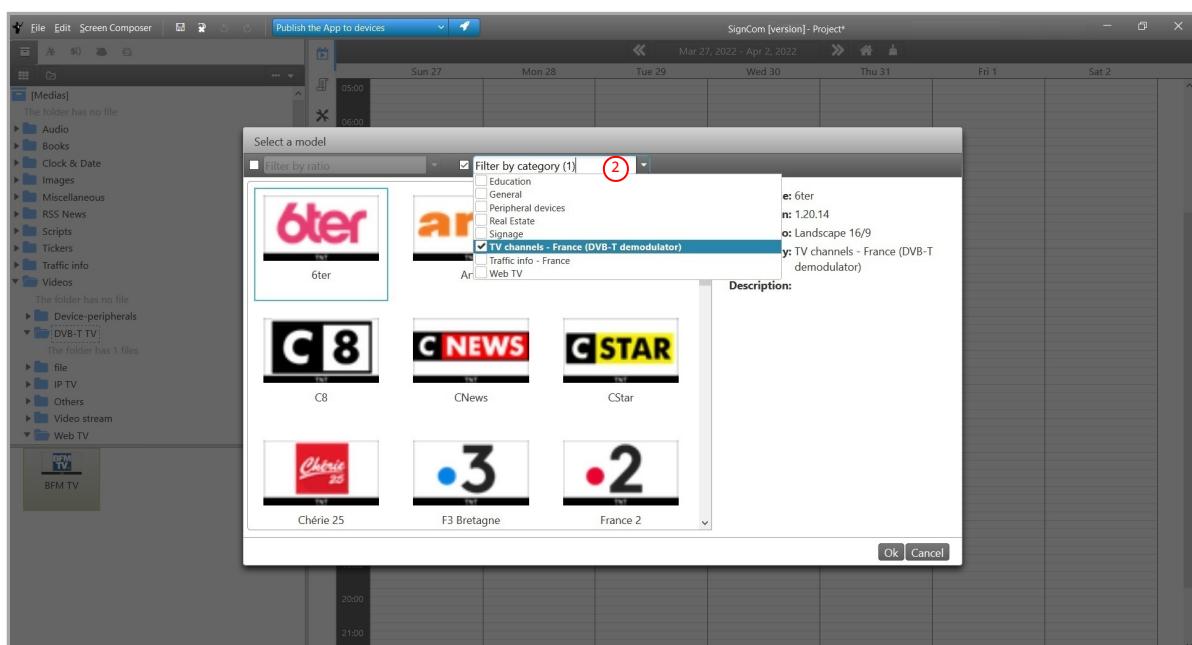
¹ By default, the library of URI models in INNES Screen Composer G4 is empty. The TV channels France URI model pack needs to be installed first in your repository to support DVB-T URI creation from DVB-T URI models. For further information, refer to the chapter § [Screen Composer > Preferences > Repositories](#).

To create a French channel DVB-T URI, it is advised to use a TV channels- France (DVB-T demodulator) URI model.

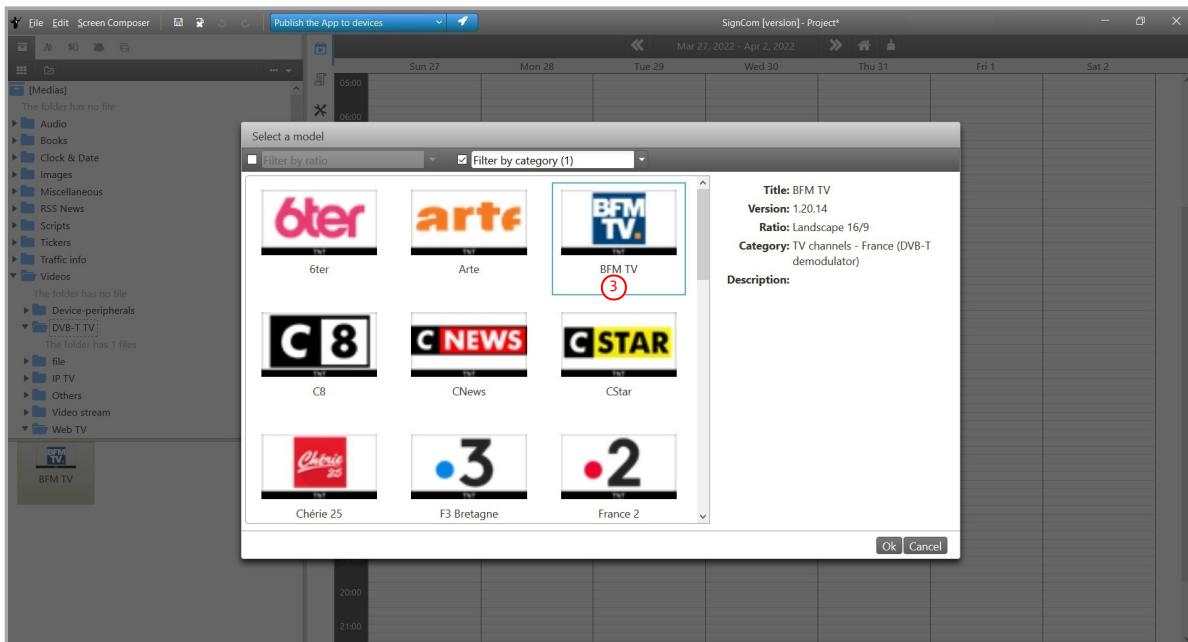
Select a folder and right click on it to display its contextual menu. Select the Create a file item then click on the Create from a model (1) item.



In the Filter by category (2) drop down list, select TV channels- France (DVB-T demodulator).

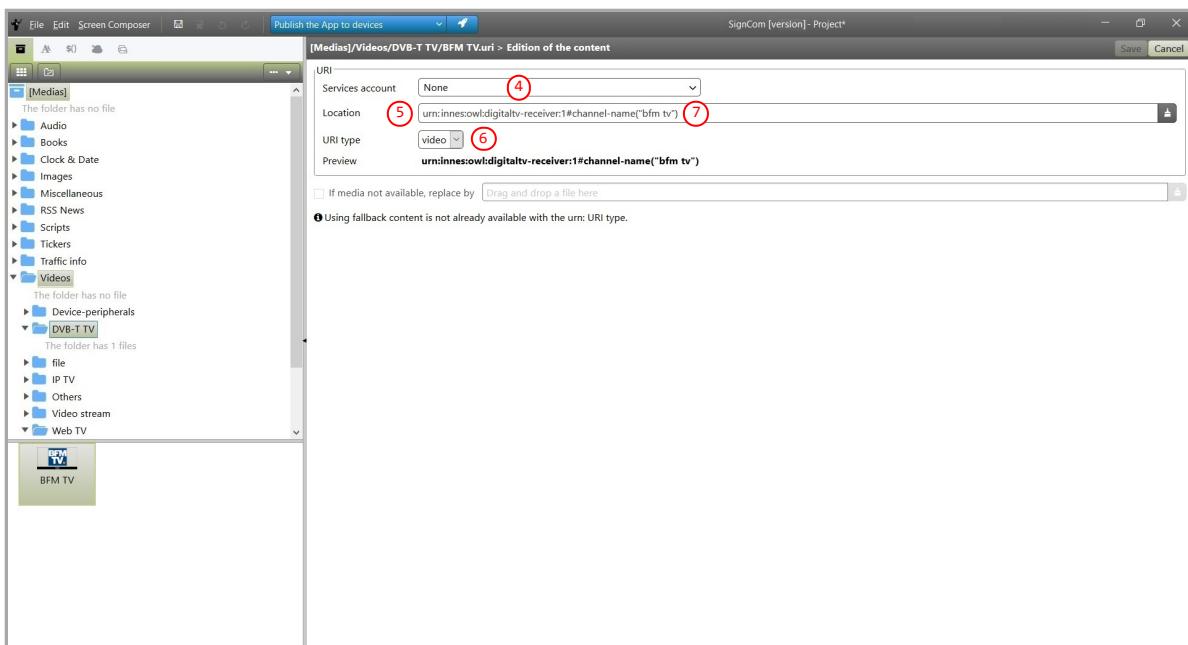


In the list of TV channels- France (DVB-T demodulator) URI models, select for example, the BFM TV channel name (3). Click on the ok button and enter a label for the URI .



For countries other than France, to create a DVB-T URI supporting other DVB-T channels names, create a DVB-T URI manually:

- by keeping the *None* services account (4),
- selecting the *urn:* protocol and matching the following URL pattern: *urn:innes:owl:digitaltv-receiver:1#channel-name("<channel_name>")* (5),
- choosing the *video* value for URI type (6).



DVB-T URI are suitable for devices supporting a DVB-T tuner card. The <channel_name> (7) is the exact service name detected by your DVB-T tuner card after the channel scanning has been launched from the device's Web user interface (Configuration > Extension > DVB card).

Web TV URI

To create a Web TV URI :

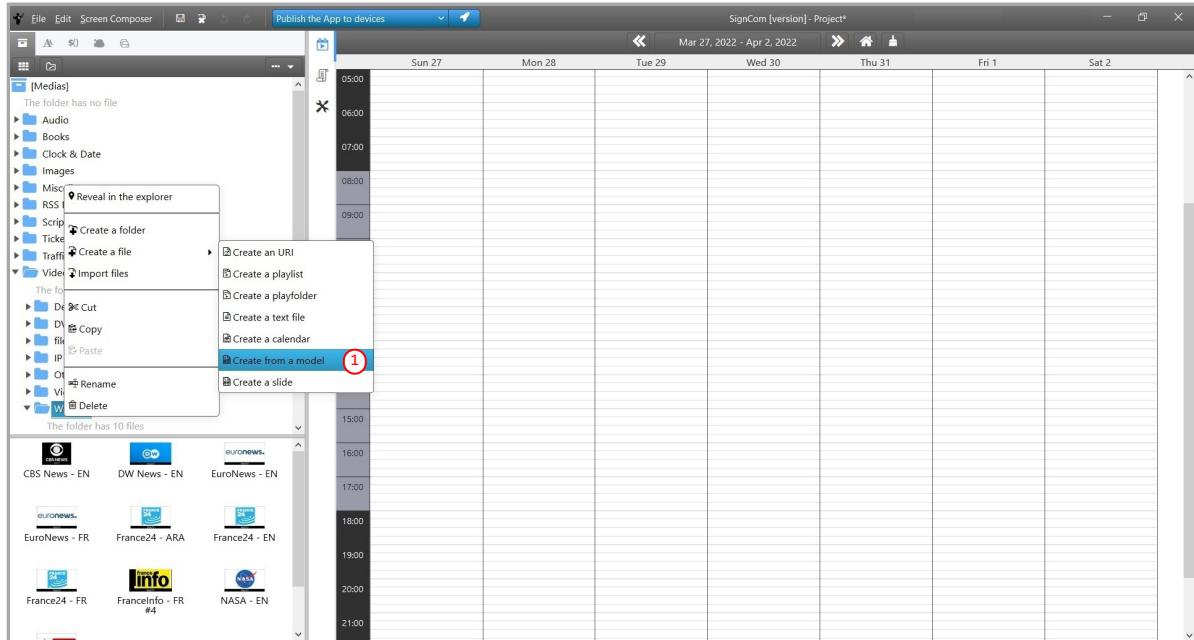
- create an URI from a Web TV URI models¹,
- create an URI by entering the Web TV URL manually with its appropriate https:// or http:// protocol,
- create an URI by pasting the Web TV URL with its protocol.

Web TV URI are not supported on Gekkota_OS 3.YY.ZZ devices.

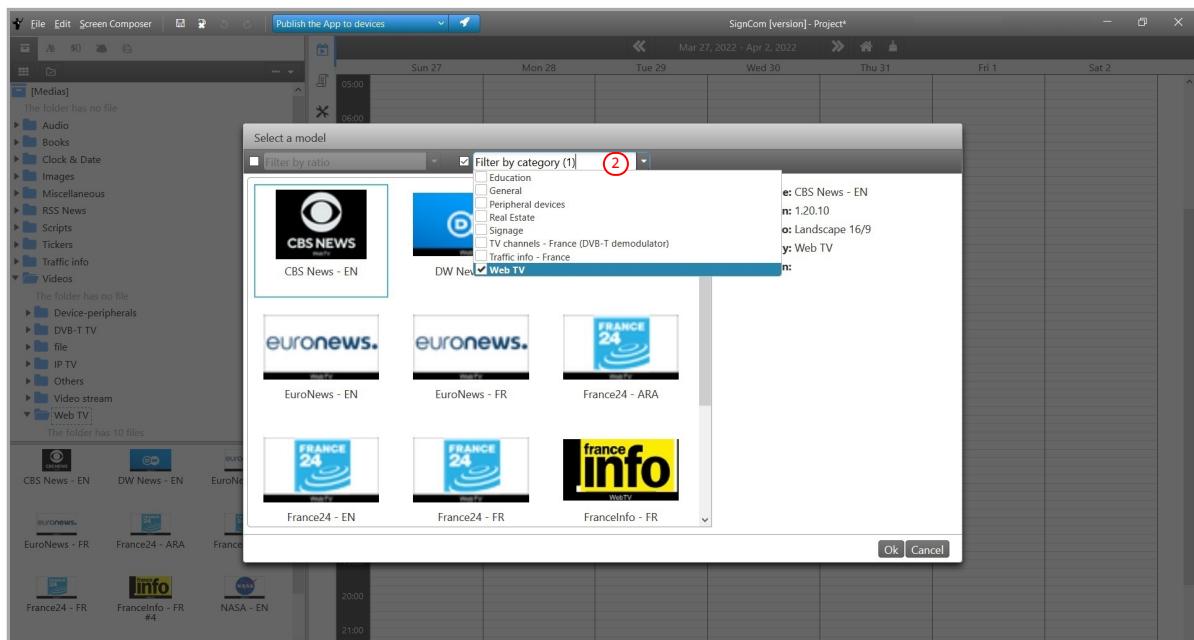
¹ By default, the library of URI models in INNES Screen Composer G4 is empty. The Web TV channels URI model pack needs to be installed first in your repository to support the Web TV URI creation from the Web TV URI models. For further information, refer to the chapter § [Screen Composer > Preferences > Repositories](#).

To create a Web TV URI, we recommend using a Web TV URI template, which lists Web TVs that accept full-screen display, such as France24 and Euronews.

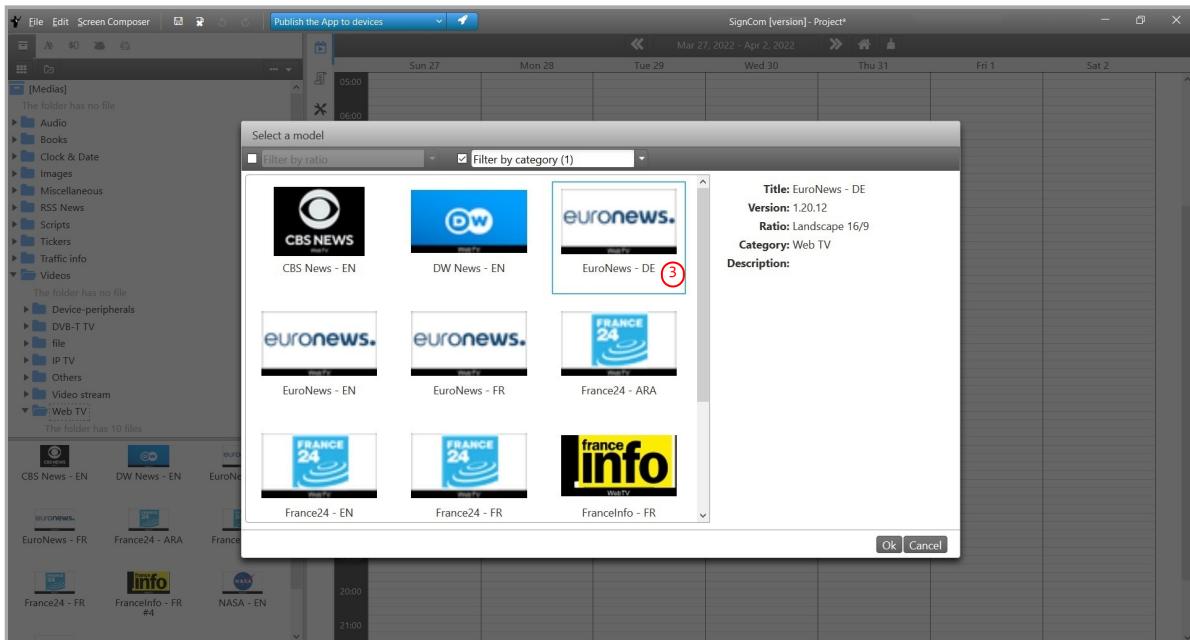
Select a folder and right click on it to display its contextual menu. Select the Create a file item then click on the Create from a model (1) item.



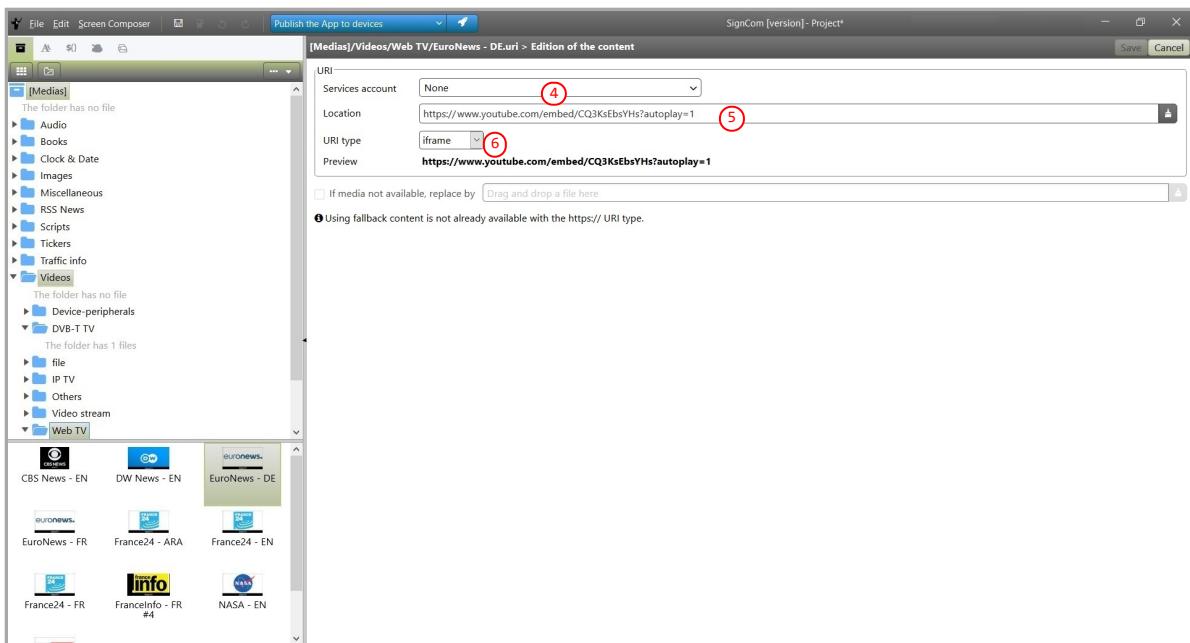
In the Filter by category (2) drop down list, select Web TV.



In the list of `Web TV URI` models, select for example, the `Euronews - DE` channel name (3). Click on the `ok` button and enter a label for the `URI`.



It is possible to create a `URI` based on your own `Web TV` URL. In this case, choose the `None` services account (4), paste the appropriate URL (5) in the location input of the `URI` form and choose the `iframe` value for `URI type` (6).



Tip: The `Web TV URI` models use video ID coming from YouTube. The YouTube video ID may have expired since the `Web TV URI` models have been created. To workaround, find out the up to date video ID by looking for the video URL with the YouTube search bar by entering the keywords "channel name" + "language" + "live" (ex: `euronews + english + live`). Once found, copy only the video ID (e.g. `sPgqEHsONK8`) of this URL (e.g.: `https://www.youtube.com/watch?v=sPgqEHsONK8`) and replace the previous ID by pasting it in the `Location` input in the `URI` form (e.g.: `https://www.youtube.com/embed/sPgqEHsONK8?autoplay=1`).

Device peripherals URI

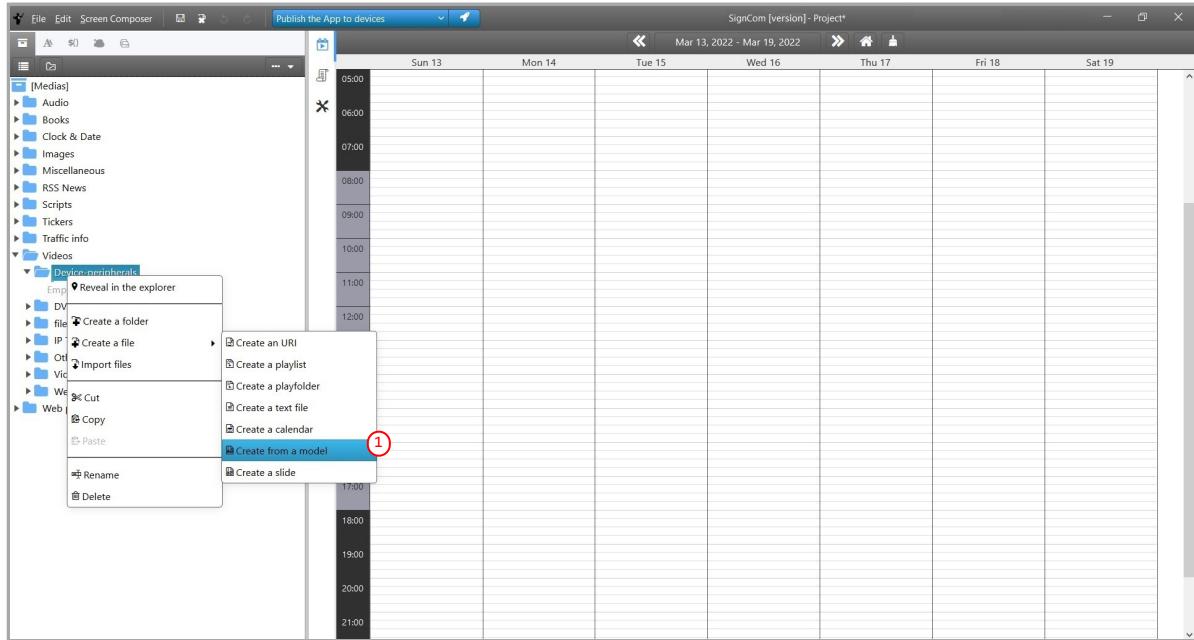
The `Device peripherals URI` handles specific device peripherals like `Audio video input` (for example to handle the HDMI-input of your device).

To create an `Audio video input URI` handling the HDMI-input on your device:

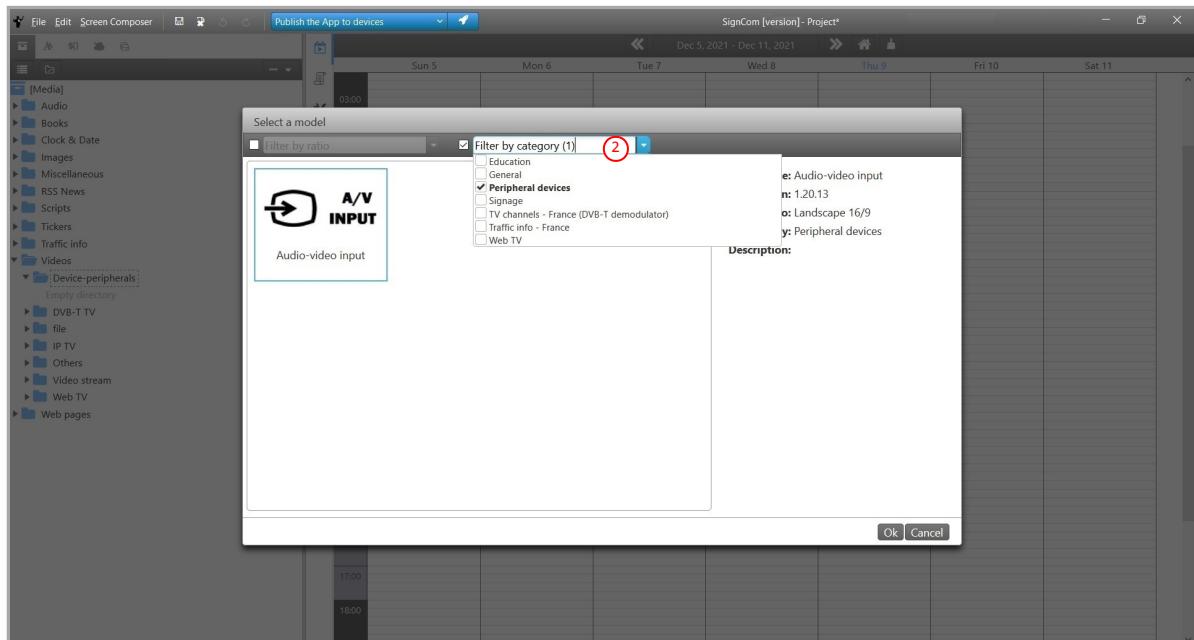
- create an `URI` from a `Audio video input URI` models¹,
- create an `URI` by entering the `Audio video input URL` manually with its appropriate protocol,
- create an `URI` by pasting the `Audio video input URL` with its protocol.

¹ By default, the library of `URI` models in `INNES Screen Composer G4` is empty. The `Device peripherals URI` model pack needs to be installed first in your repository to support `Audio video input URI` creation from `Device peripherals URI` models. For further information, refer to the chapter § [Screen Composer > Preferences > Repositories](#).

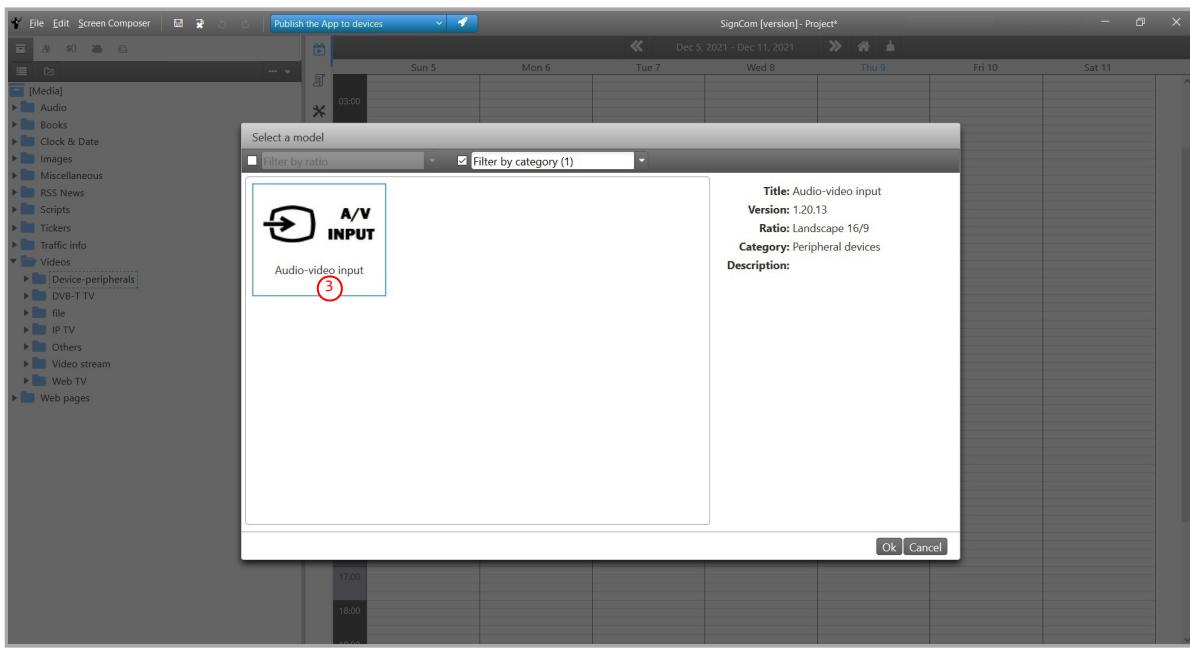
Select a folder and right click on it to display its contextual menu. Select the `Create a file` item then click on the `Create from a model` (1) item.



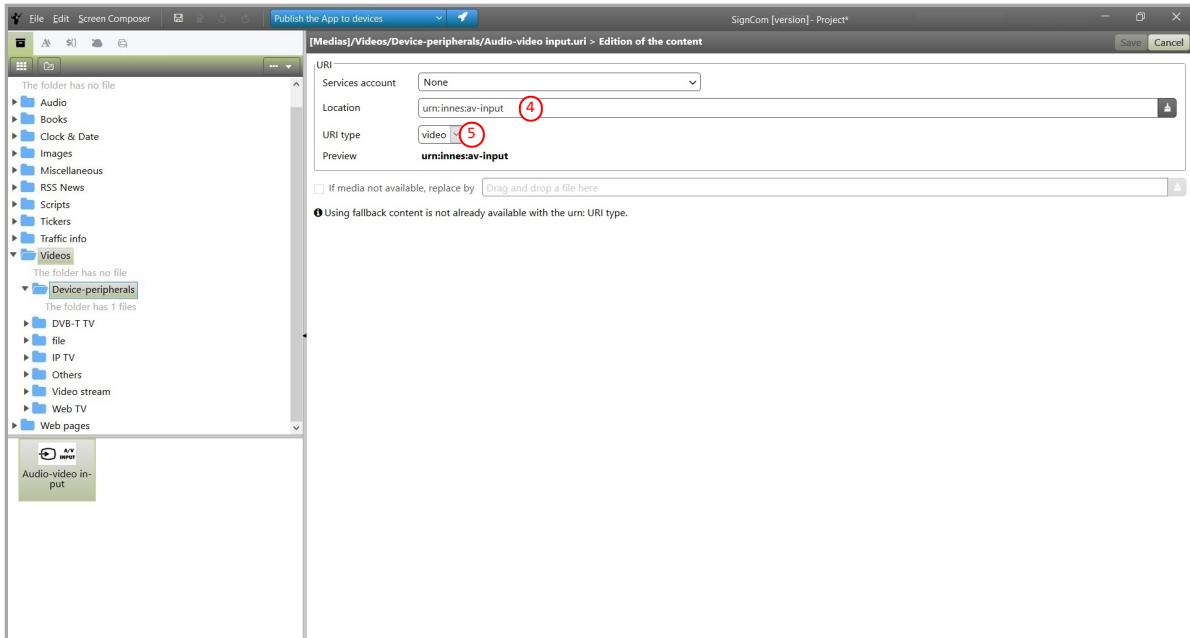
In the `Filter by category` (2) drop down list, select `Peripheral devices`.



Select the `Audio video input` (3) URI model. Click on the `ok` button and enter a label for the `URI`.



The location `input` value is `urn:innes:av-input` (4) and its `URI type` value must be `video` (5).



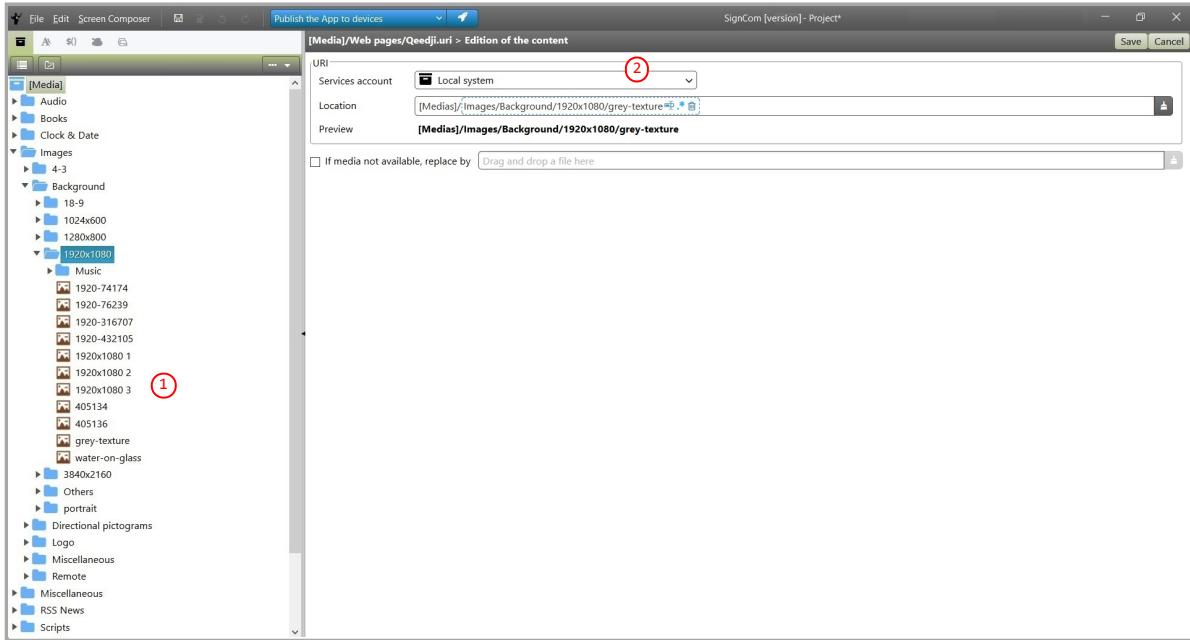
Shortcut URI

A shortcut to a local media of your library can be done with an `URI`.

Select the `Local system` services account.

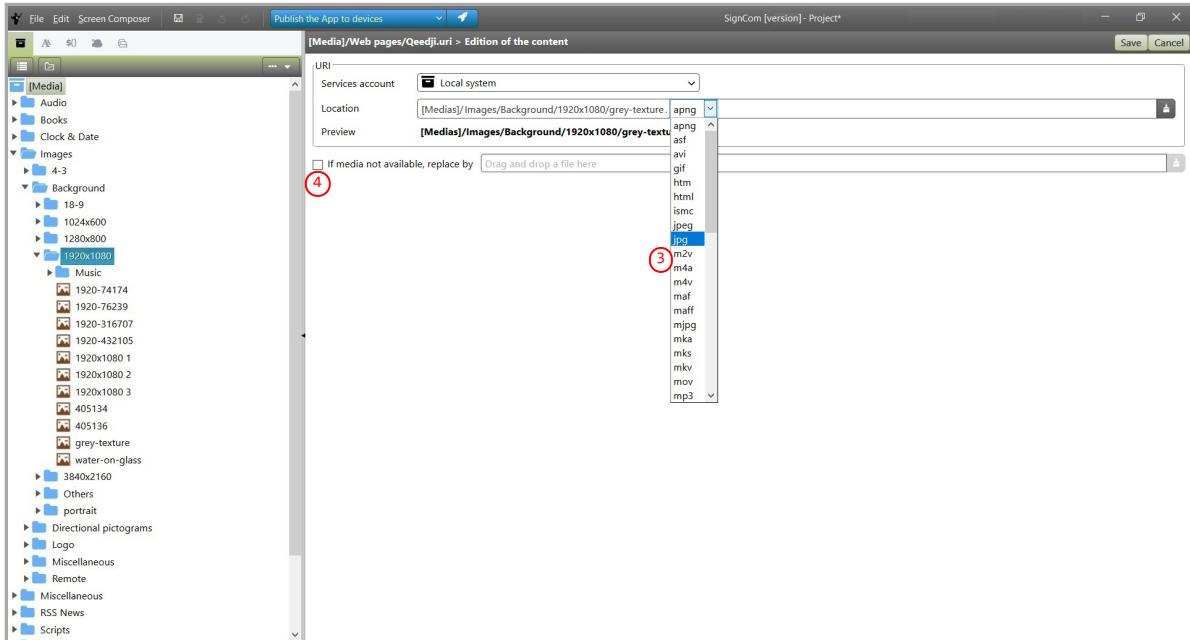
At the `location`, select the `[Medias]/` scheme.

To fill the `location` input (1) of your `URI` form, it is advised to drop a media from the `Media` tab to the `location` input (1) of your `URI` form. If the `Local system` services account is properly selected and the `[Medias]/` scheme present, you can enter also the file name URL manually.



If no valid extension is detected at the end of the `location` input value, the `URI` cannot be saved. So fly the mouse pointer over the last element of your path and click on the (2) button to select the extension among the following one (3): `.apng`, `.ASF`, `.avi`, `.gif`, `.htm`, `.HTML`, `.ismc`, `.jpg`, `.jpeg`, `.m2v`, `.m4a`, `.m4v`, `.maf`, `.maff`, `.mjgp`, `.mka`, `.mks`, `.mkv`, `.mov`, `.mp3`, `.mp4`, `.mpd`, `.mpg`, `.mpeg`, `.mxaf`, `.odt`, `.odp`, `.oga`, `.ogg`, `.ogv`, `.pdf`, `.png`, `.pps`, `.ppsx`, `.ppt`, `.pptx`, `.ps`, `.sdp`, `.svg`, `.ts`, `.vob`, `.webm`, `.webp`, `.wgt`, `.wma`, `.wmv`.

The `URI` of `.uri` files, the `URI` of `.playlist` files and the `URI` of `.playfolder` files are not supported.



Fallback

For the `URI` having the `[Medias]/` URL scheme, checking the `If media not available, replace by` (4) fallback checkbox allows to define an alternative media to play if ever the content defined in the `URI` would not be available.

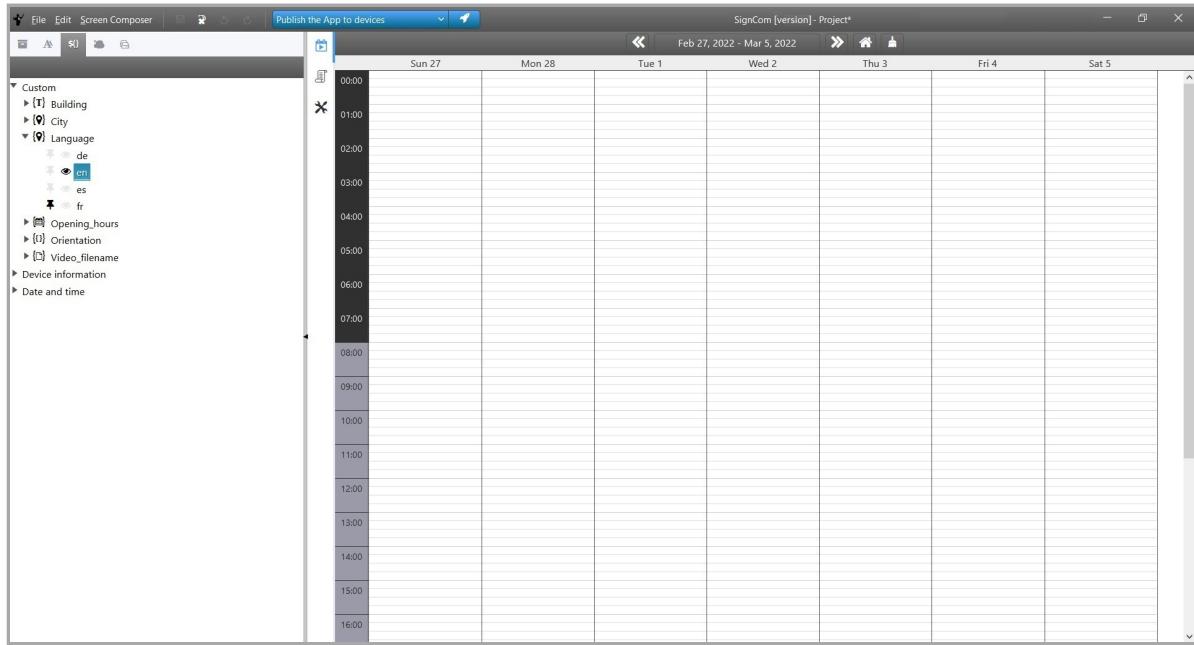
Using variables

It is possible to create an `URI` media whose content is conditionned by:

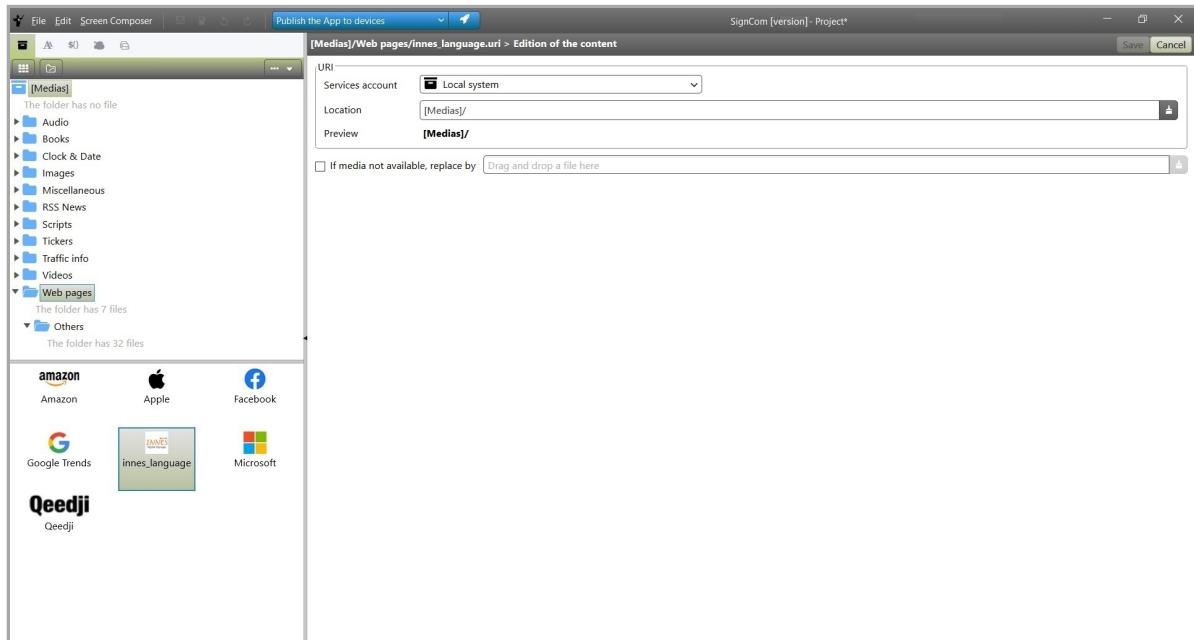
- a Custom localized variable,
- a Custom relative URL variable,
- a Device information variable,
- a Date and time variable.

In the following example, it is considered that the `URI` must be conditionned by the custom localized variable name `Language` which has four values:

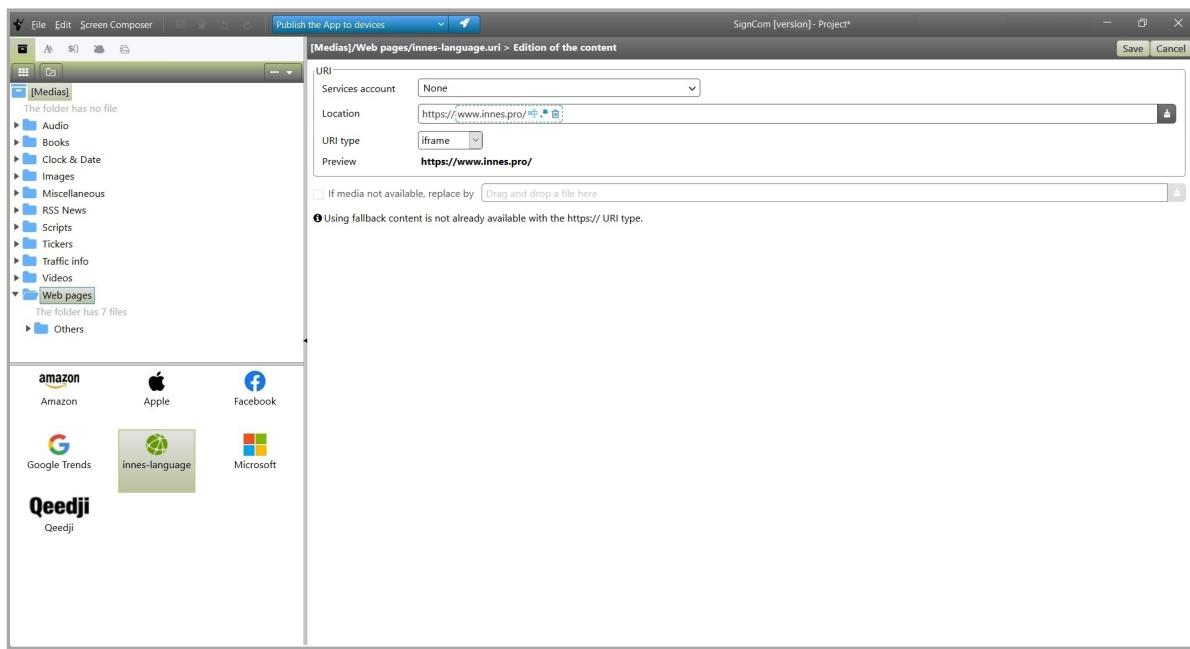
- `de`,
- `en`,
- `es`,
- `fr`.



To create an `URI`, select an appropriate folder and right click on it to display its contextual menu. Select the Create a file item then click on the Create an `URI` (1) item. Enter a label for the `URI` (e.g. `innes_language`) and click on the `ok` button.

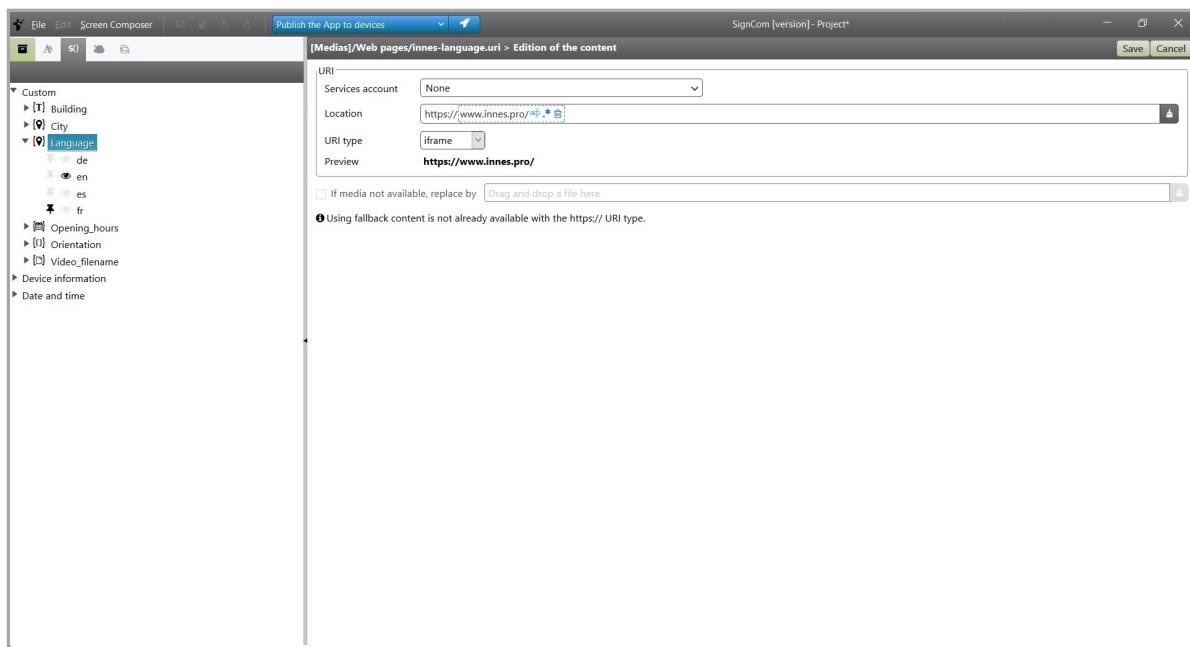


Select the **None** services account. In the **location** input, paste a Web page URL supporting for example several languages thanks to the URL suffix **/en, /fr, /de** (e.g. <https://www.innes.pro/>).

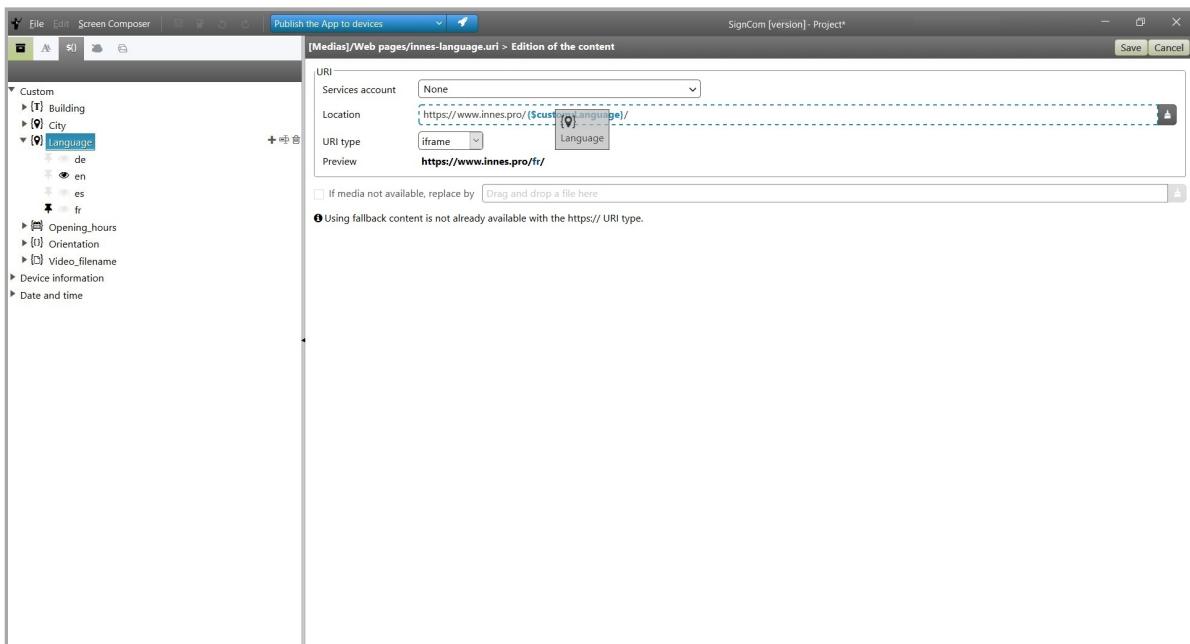


Select the **\$0** Variables tab.

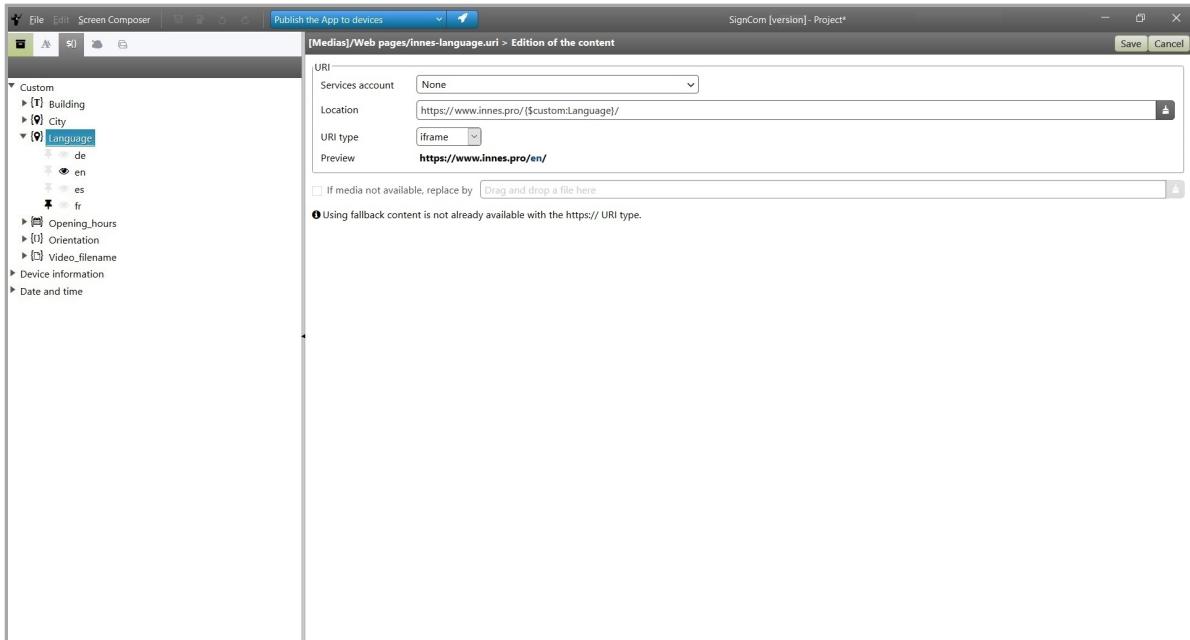
With your mouse, fly over the end of the **location** input of the **URI** to make appear the **Edit** blue button and click on it.



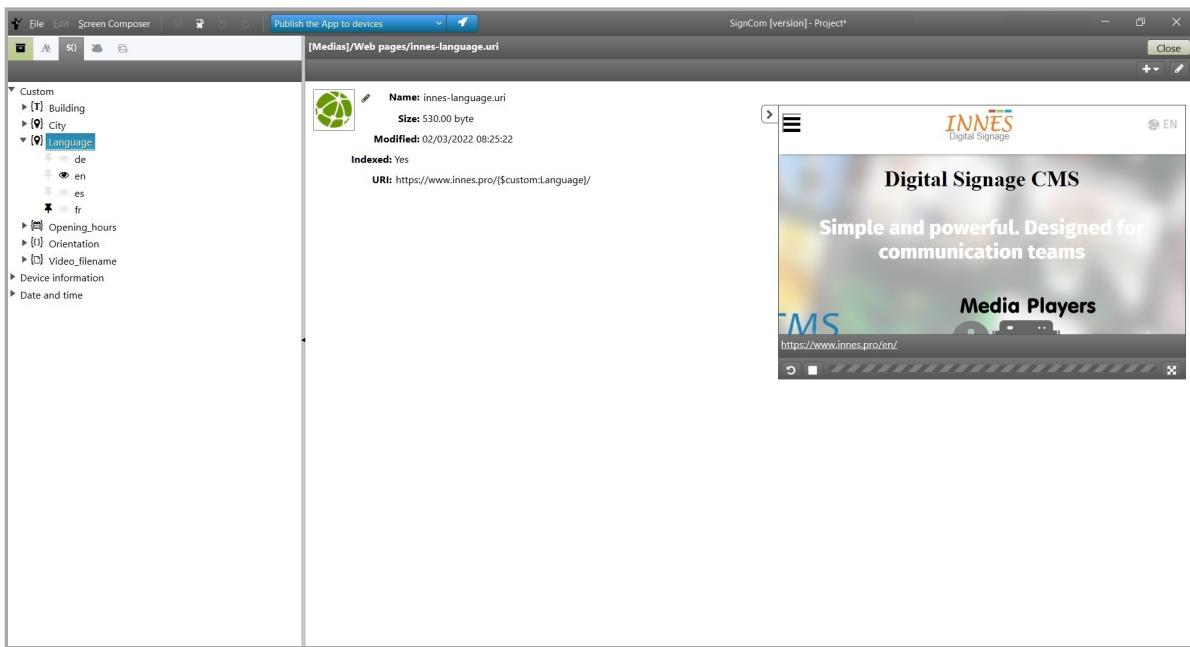
Drop the wished `custom` localized variable name (e.g. `Language`) in the `location` input.



The preview URL of the `URI` is shown just below by using the variable value for preview.



Save the `URI` (e.g. `innes-language`). Launch the preview of the `innes-language` `URI`. The content of the `https://www.innes.pro/en/` is played



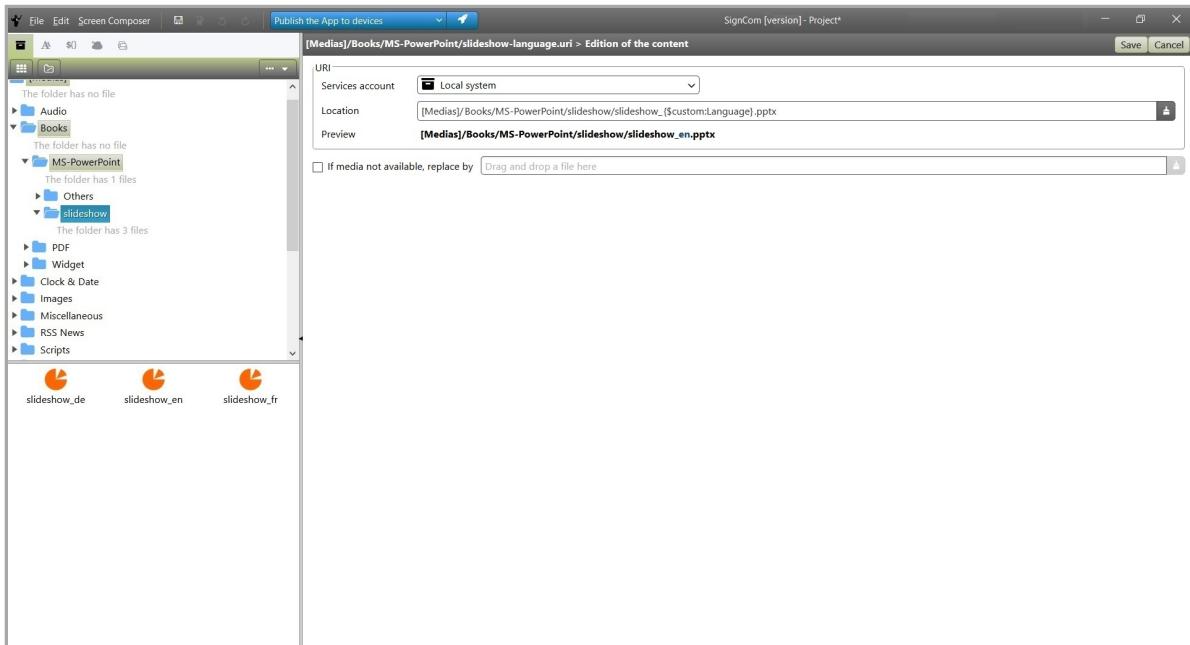
Depending on the `custom` localized variable `Language` value published on the devices, the `URI` `innes-language` will play the content of:

- `https://www.innes.pro/fr/`,
- `https://www.innes.pro/en/` or
- `https://www.innes.pro/de/`.

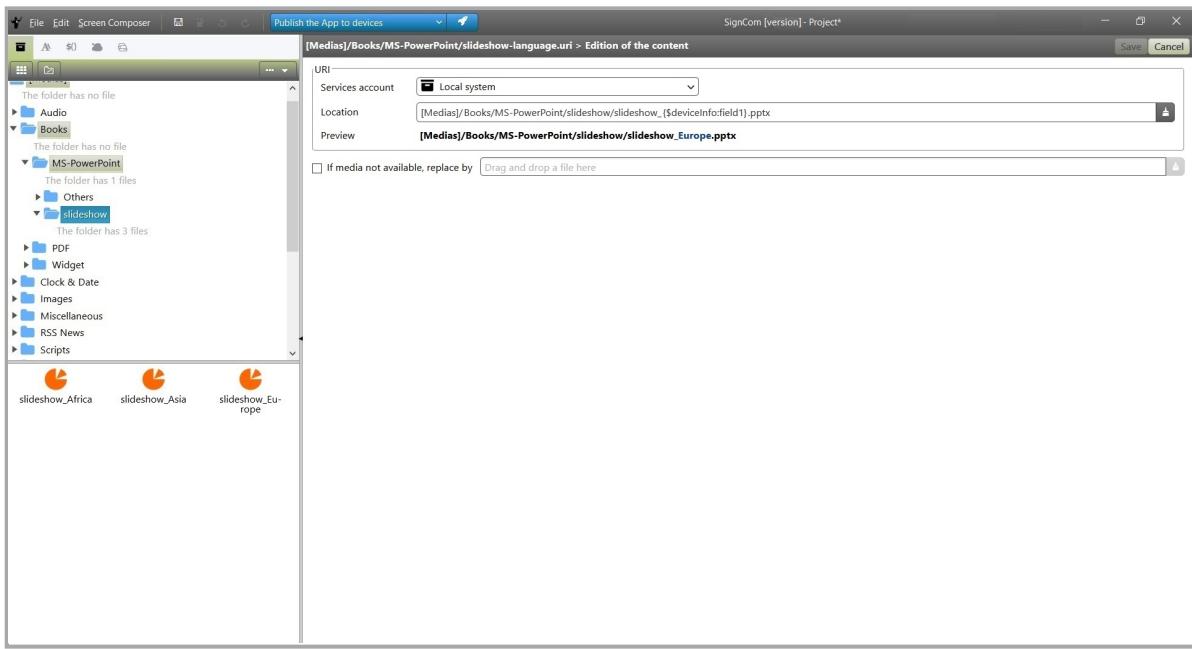
When using a `Custom` localized variable name, the `{$custom:<variable_name>}` is pasted in the `location` input of the `URI`.

Tip Using a personalised variable of type `relative URL` is done in the same way as processing a personalised variable of type `Localized`. If, for example, the `relative URL` variable `qeedji` has the values `www.innes.pro/en` and `www.innes.pro/fr`, choose the appropriate scheme (e.g. `https://`) in the `Location` input of the `URI`, then drag the name of the relative variable from the variable library to the `Location` input.

This is another example of `URI` allowing to play a MS-PowerPoint presentation depending on the `custom` localized variable `Language`.

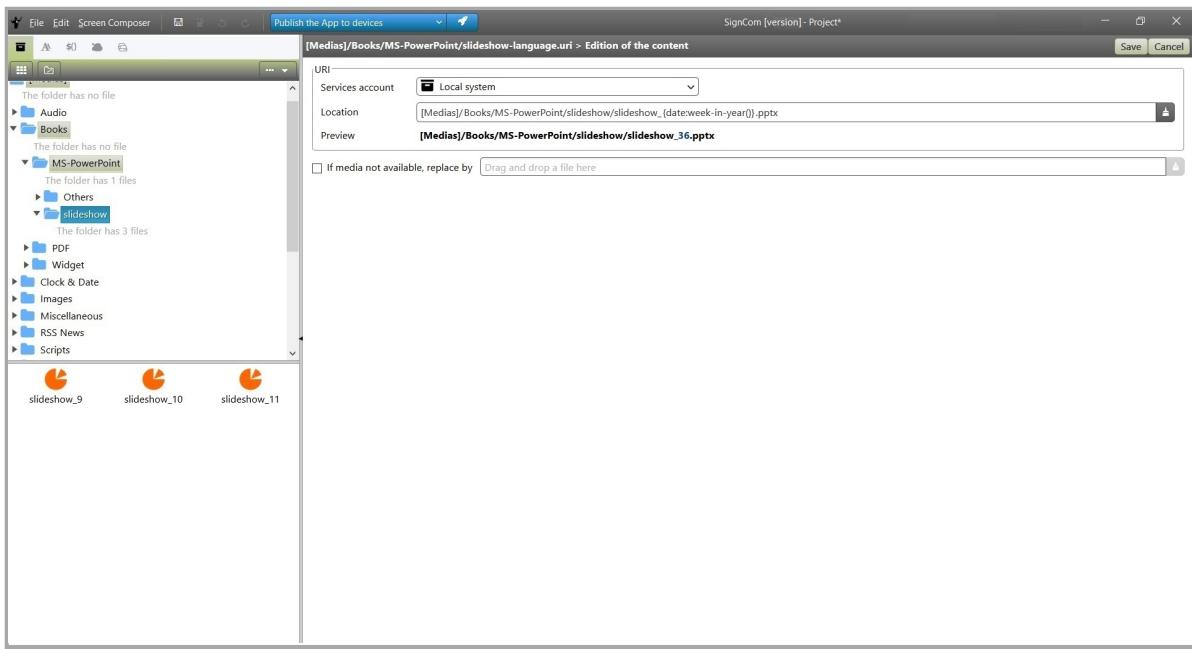


When using a **Device Information** variable name, the `${deviceInfo:<variable_name>}` is pasted in the **location** input of the **URI** (e.g. `${deviceInfo:field1}`).



⚠ Publishing URI using Device information variable name requires to add manually the required target files or target folders in the manifest.
For further information, refer to the chapter § [SignCom > Manifest](#).

When using a **Date and time** variable name, the `{date:<variable_name>}` is pasted in the **location** input of the **URI** (e.g. `{date:week-in-year()}`).



⚠ Publishing URI using Date and time variable name requires to add the manually the required target files or target folders into the manifest.
For further information, refer to the chapter § [SignCom > Manifest](#).

URI with credentials

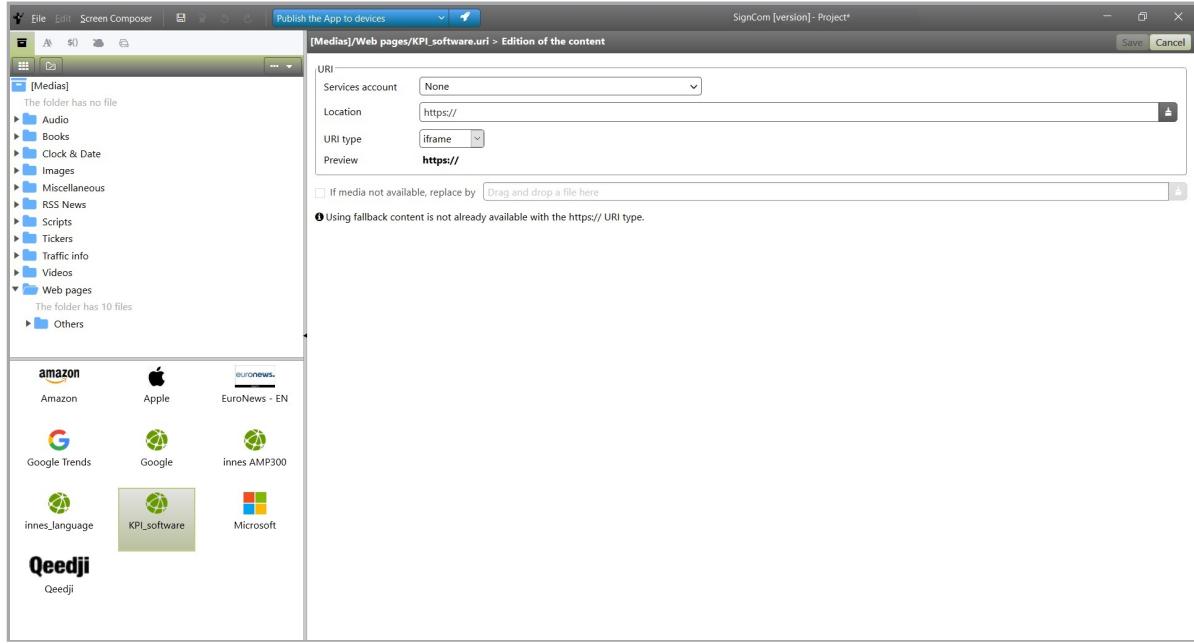
To create an `URI` with credentials, you must have created first a `Simple Web server services account` either with:

- an `Identifier/Password for a basic authentication` or
- an `Identifier/Password for a webpage form`.

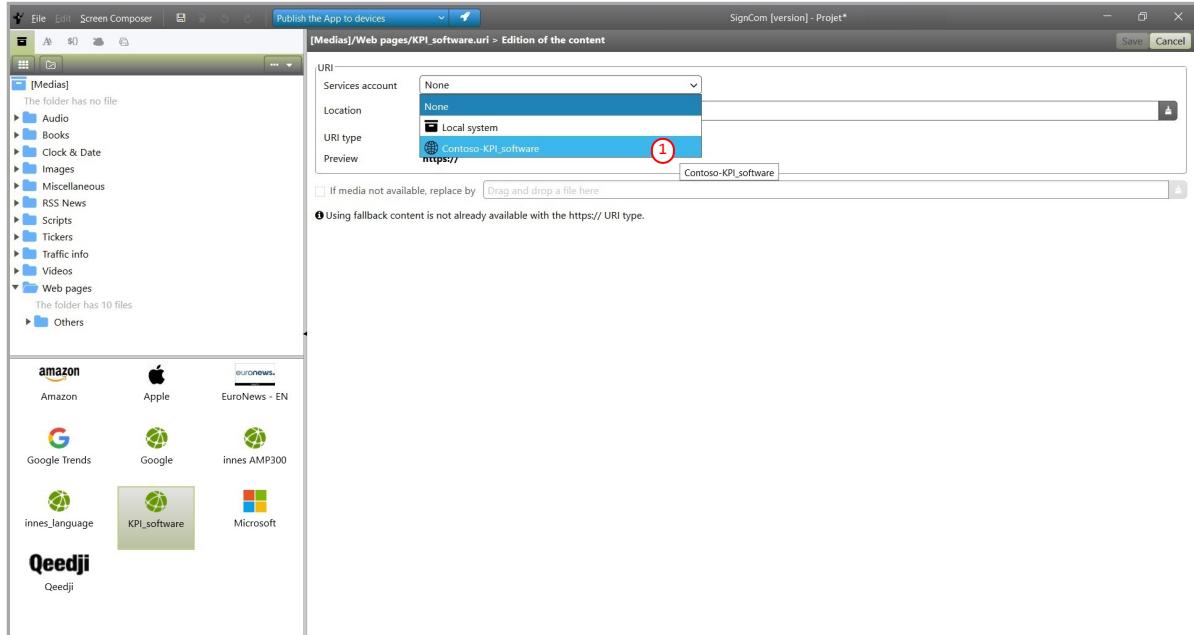
The first one is used when the credentials inputs must be filled automatically inside a identifier/password popup supporting a basic authentication. The second one is used when the credentials inputs are located directly inside the Web page.

It is considered in the next steps that an appropriate `Simple web server services accounts` has been created with the right credential type. For further information, refer to the chapter § [Services accounts](#).

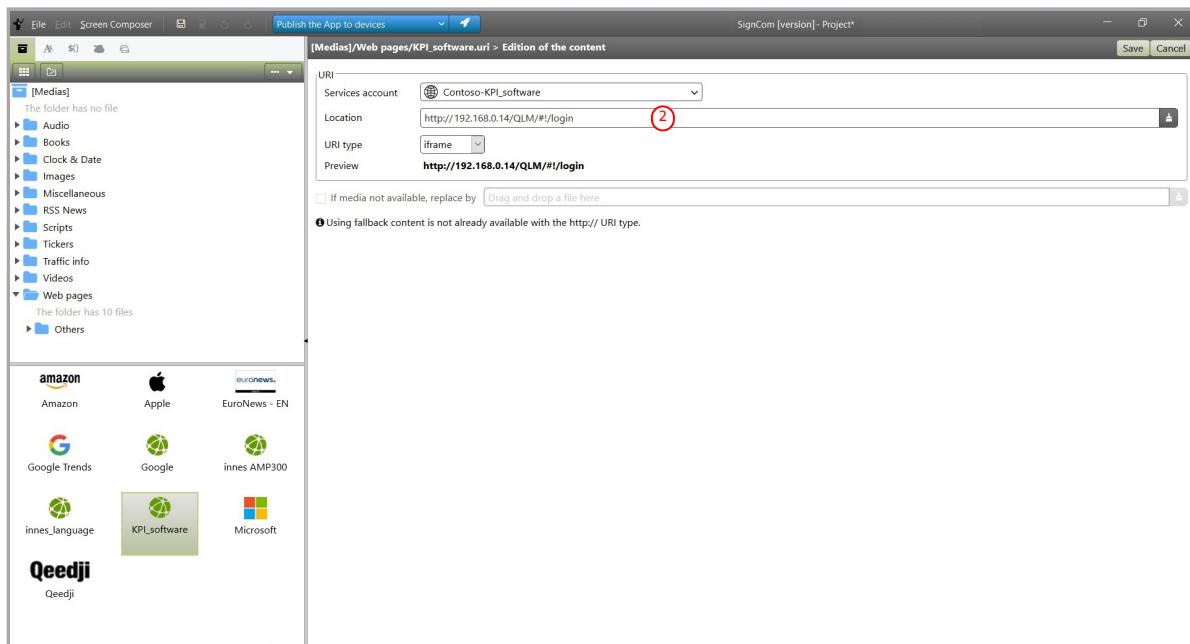
To create an `URI`, select an appropriate folder and right click on it to display its contextual menu. Select the `Create a file` item then click on the `Create an URI` item. Enter a label for the `URI` (e.g. `KPI_software`) and click on the `ok` button.



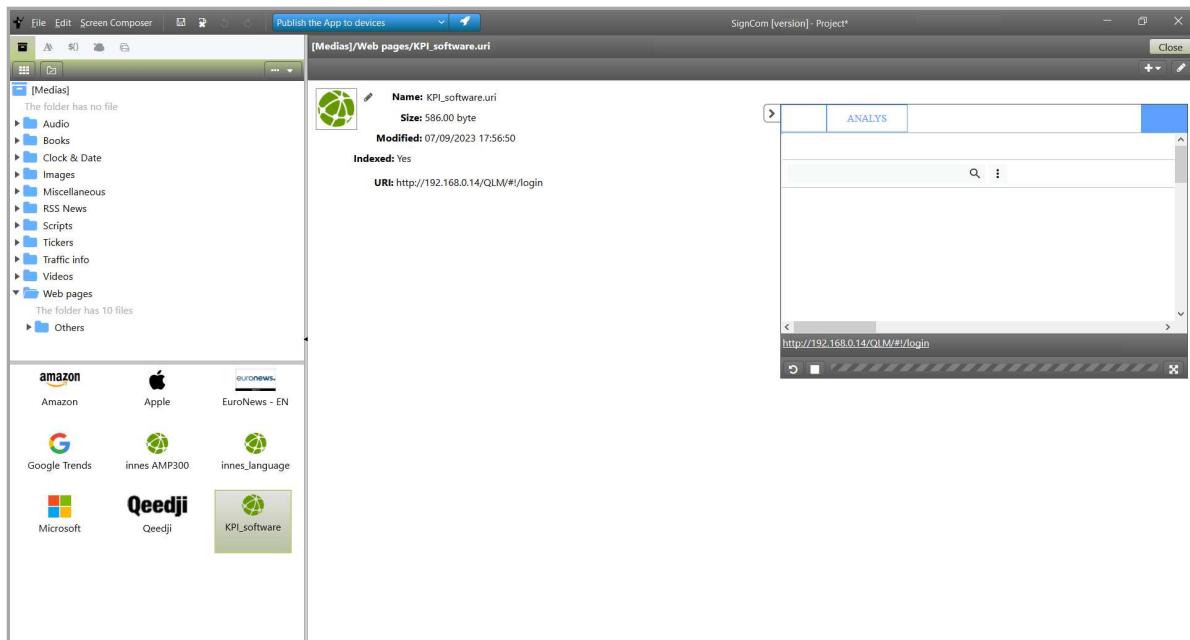
Select the `Simple Web server services account` you have created before (e.g. `Contoso-KPI_software` ①).



And paste the URL of your Web page in the `location` input (2) of the `URI` form.



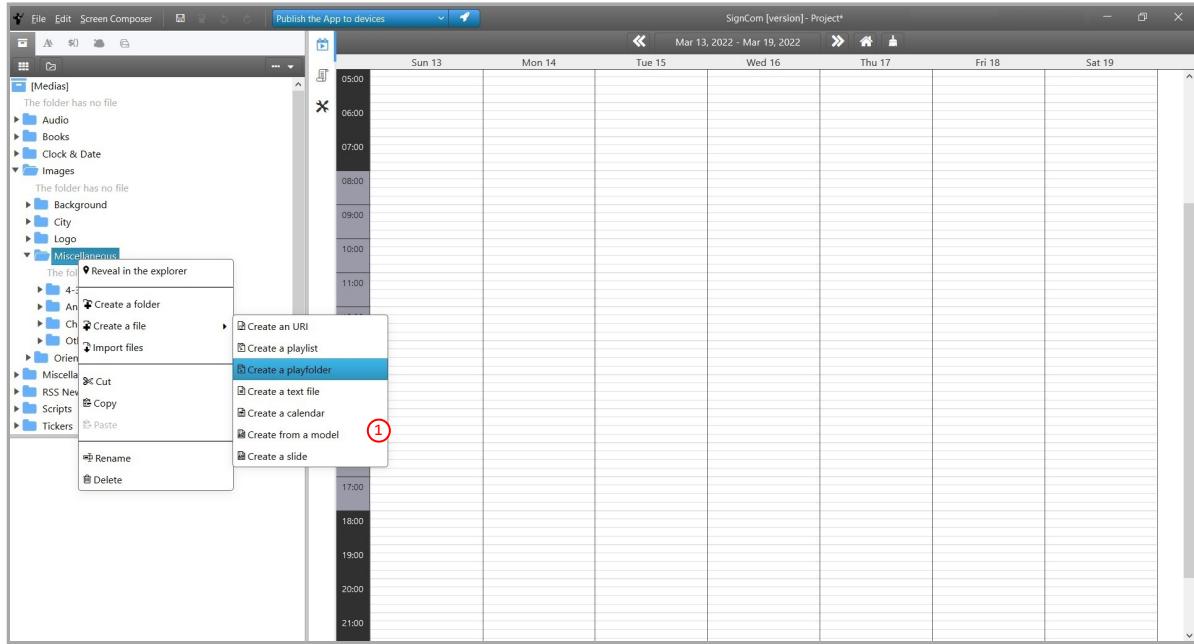
Save the form. You must be able to preview your URI in the pane.



3.2.3 Medias > Playfolder

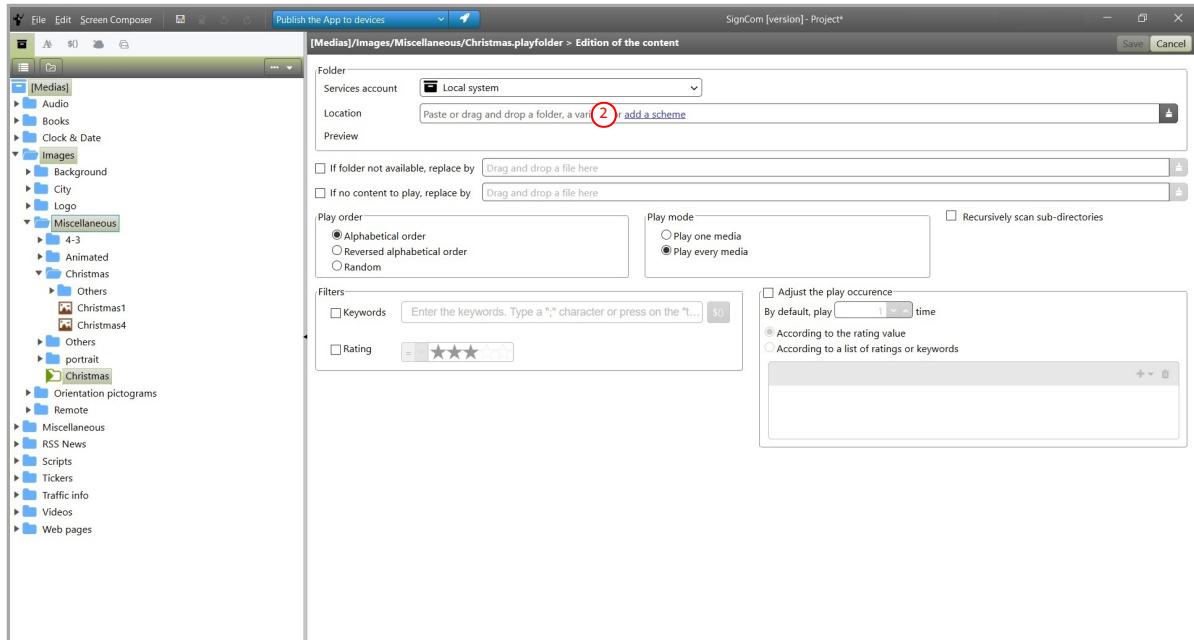
A playfolder media has a `.playfolder` file extension. It allows to create a media which is playing some others medias included in a folder and its sub-directories by following some playing rules.

To create a playfolder, select an appropriate folder and right click on it to display its contextual menu. Select the **Create a playfolder** item then click on the **OK** button.



Enter a name for the `.playfolder` file (e.g. *Christmas*) then click on the **OK** button.

To finalize the creation of your playfolder, enter the folder path name of an existing folder of your medias library in the `location` input **②** of the playfolder form.

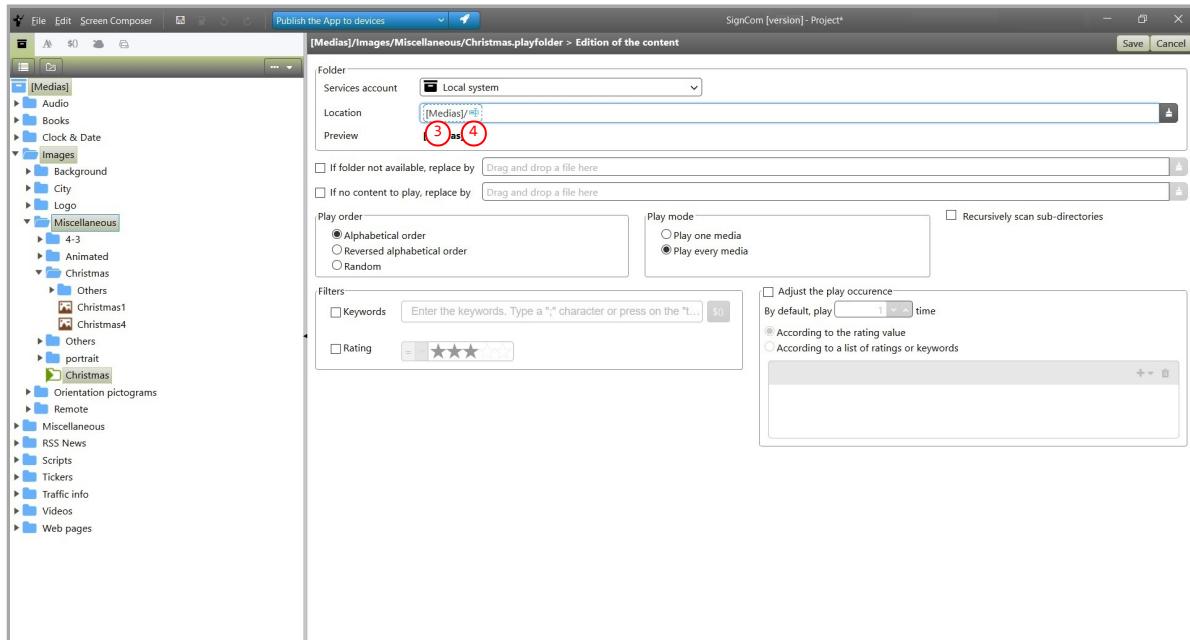


The `Service account` value in the drop down list of any new `playfolder` is by default `Local system`, meaning that the playfolder is by default configured to plays medias that are hosted in the medias repository.

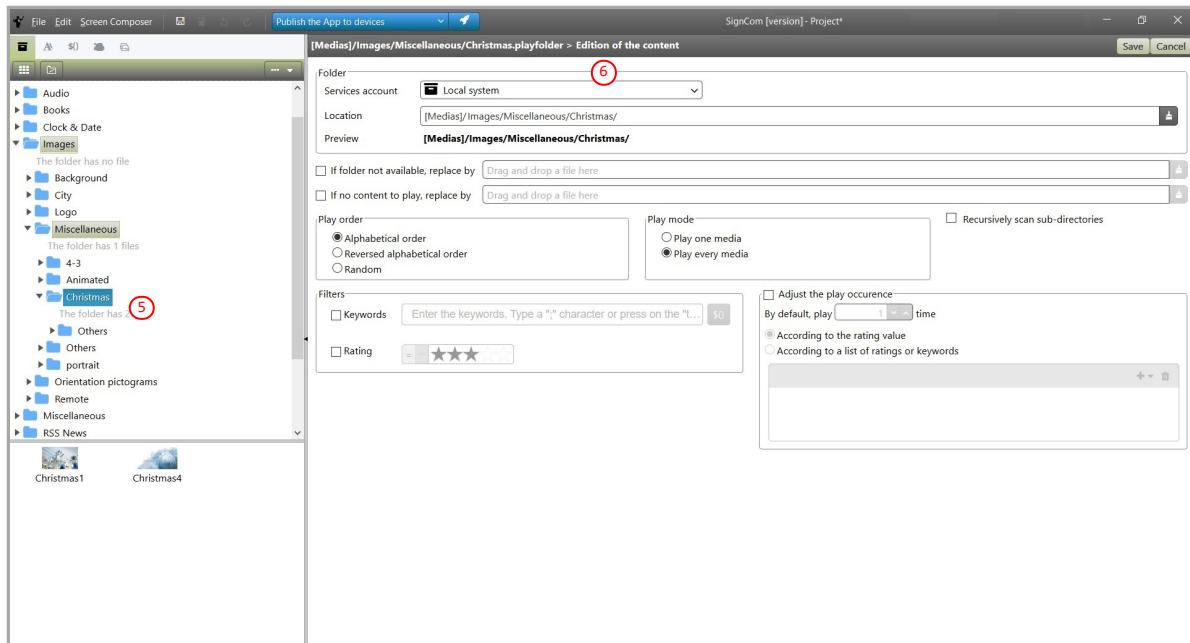
Enter the path of the folder manually:

- click on the add a scheme **②** link,
- select the [Media]\ **③** scheme and click outside the input,
- fly your mouse pointer over the [Medias]/ scheme to make appear the **④** blue button, and click on it,
- then enter the rest of the folder's path elements.

☞ The path must end with a slash / character.



You can also drag'n drop a folder **⑤** from the **Media** tab to the playfolder input **⑥** (advised solution).



Click on **Save** button to save your modifications.

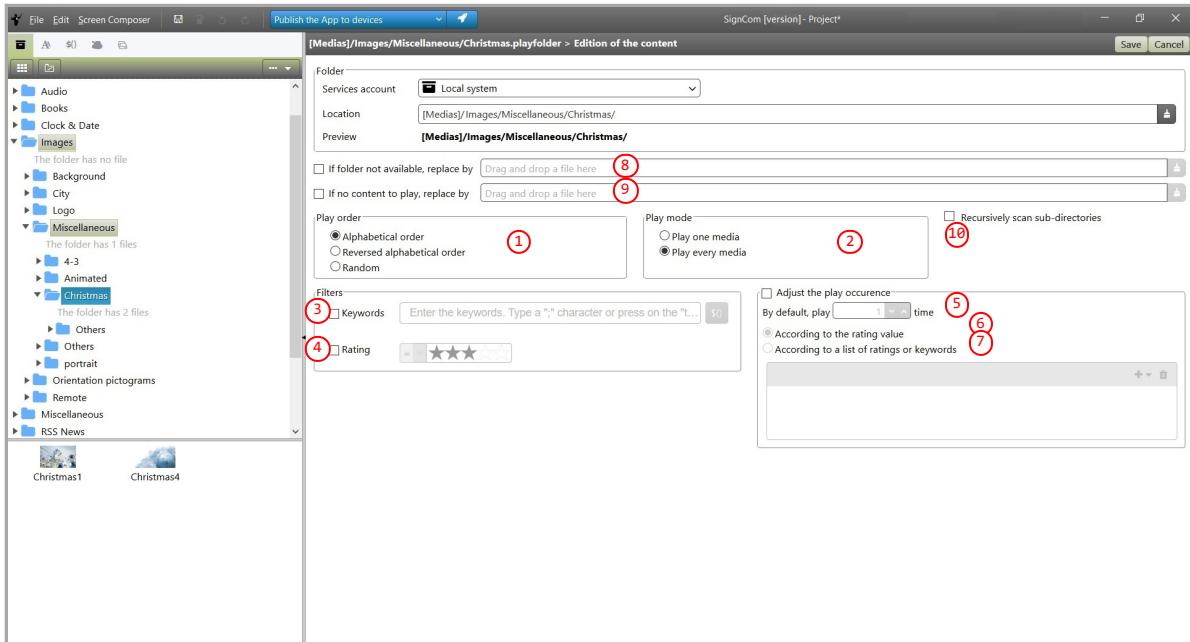
☞ The playfolder media supports for playing others `.playfolder` media, `.playList` media and `URI` media contained in the folder.

☞ The `playFolder` media supports for audio medias and visual medias. It is advised to create `playFolder` media containing either only audio medias, or only visual medias.

Playing criteria

In case your playfolder must play only some of the medias contained in your folder, it is possible to define the playing criteria of your playfolder:

- Play order ①:
 - Play the media in
 - *Alphabetical order*
 - *Reversed alphabetical order,*
 - *Randomly,*
- Play mode ②:
 - *Play one media,*
 - *Play every media,*
- Recursively scan sub-directories ⑩:
 - Play or not the contents of the subdirectories,
- Filters : allows to display only some medias based on some rating or keyword values conditions:
 - Keywords ③: allows to play only medias that have the specified keyword or set of keywords,
 - Rating ④: allows to play only medias that have some metadata rating values:
 - equal (=), lower (<) or higher (>) than a rating value
 - possible rating values: from 1 to 5 stars
- Adjust the play occurrence :
 - According to the rating value ⑥ allows to play:
 - Five times in a row, the medias with a 5-stars rating,
 - Four times in a row, the medias with a 4-stars rating,
 - Three times in a row, the medias with a 3-stars rating,
 - Two times in a row, the medias with a 2-stars rating,
 - Once, the medias with a 1-star rating,
 - Following a list of ratings or keywords³ ⑦ allows to play:
 - <n1> times in a row, the medias with a 5-stars rating,
 - <n2> times in a row, the medias with a 4-stars rating,
 - <n3> times in a row, the medias with a 3-stars rating,
 - <n4> times in a row, the medias with a 2-stars rating,
 - <n5> times in a row, the medias with a *keyword value 1*,
 - <n6> times in a row, the medias with a *keyword value 2* and a *rating value 2* , ...
 - ⑤ By default, play <n> times allows to play:
 - <n> times in a row, the medias with no special rules.
- Case at the limits:
 - ⑧ Play an alternative media when the folder is not found,
 - ⑨ Play an alternative media when the folder is empty.



¹The maximum number of characters per keyword is 260. Upper case characters of *Keywords* is changed to lower case. Anyway the filtering is working properly whatever whether the value is containing uppercase or lowercase.

²In this version, the " ; " character is not supported in the keyword metadata of JPG media.

³When both keyword and rating filters are enabled, the medias to be played must meet both conditions. When multiple keywords are specified in the filter, all keywords values must be present for a media so that it can be played.

⁴When two occurrence playing behaviours value can be applied to a same media, for example two times and five times, the upper playing occurrence is chosen for the final media playback behaviour.

Variables

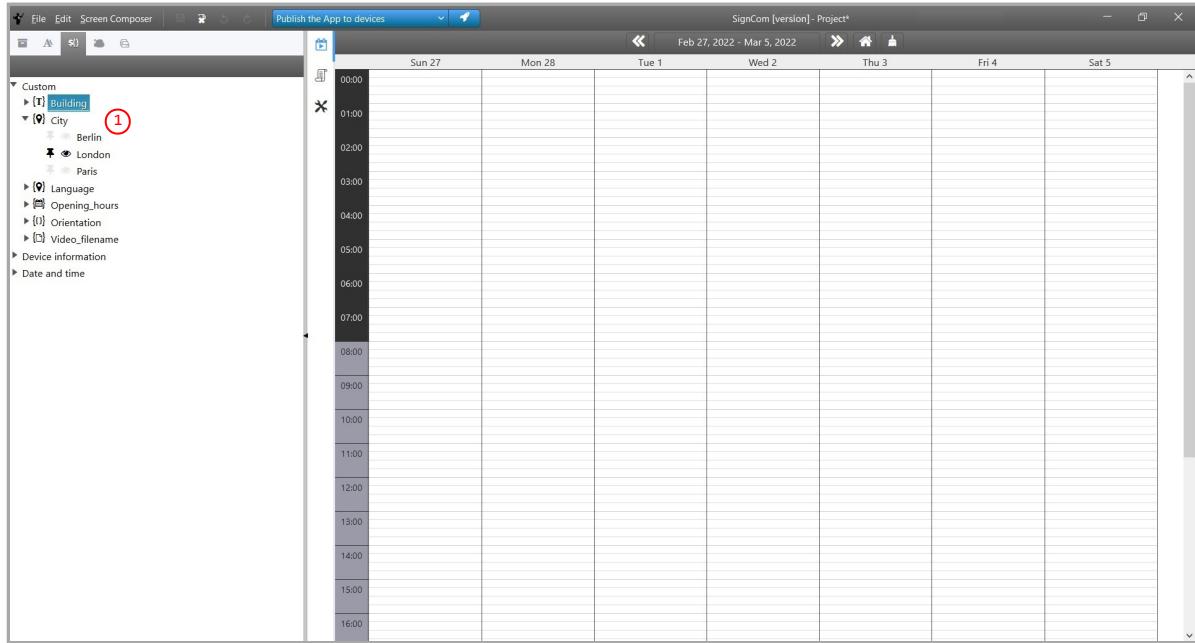
It is possible to create a playfolder media whose content is conditionned by:

- a Custom variable of type:
 - localized ,
 - relative URL (for playfolder with Microsoft 365 , CIFS and Simple web server services accounts),
- a Device information variable,
- a Date and time variable.

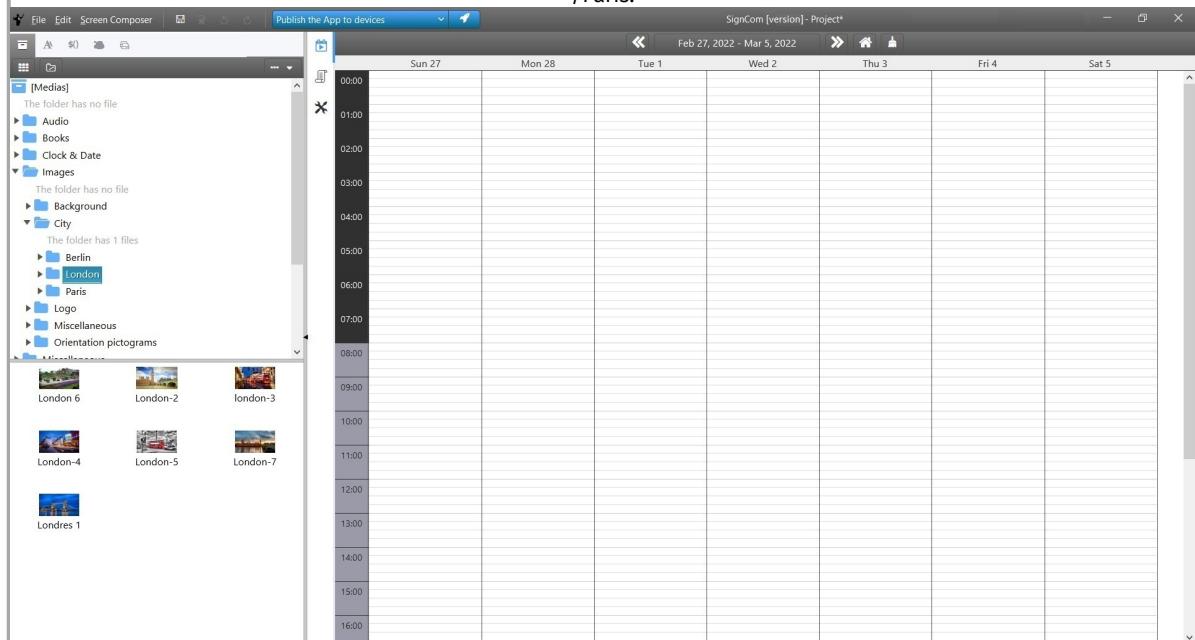
When the services account is different from the Local system, meaning Microsoft 365 services account or CIFS services account, the custom of type relative URL variable can be dropped in the playfolder location input, with smb:// scheme or https:// scheme.

In the following example, it is considered that the playfolder content must be conditionned by the custom variable name City which has three values:

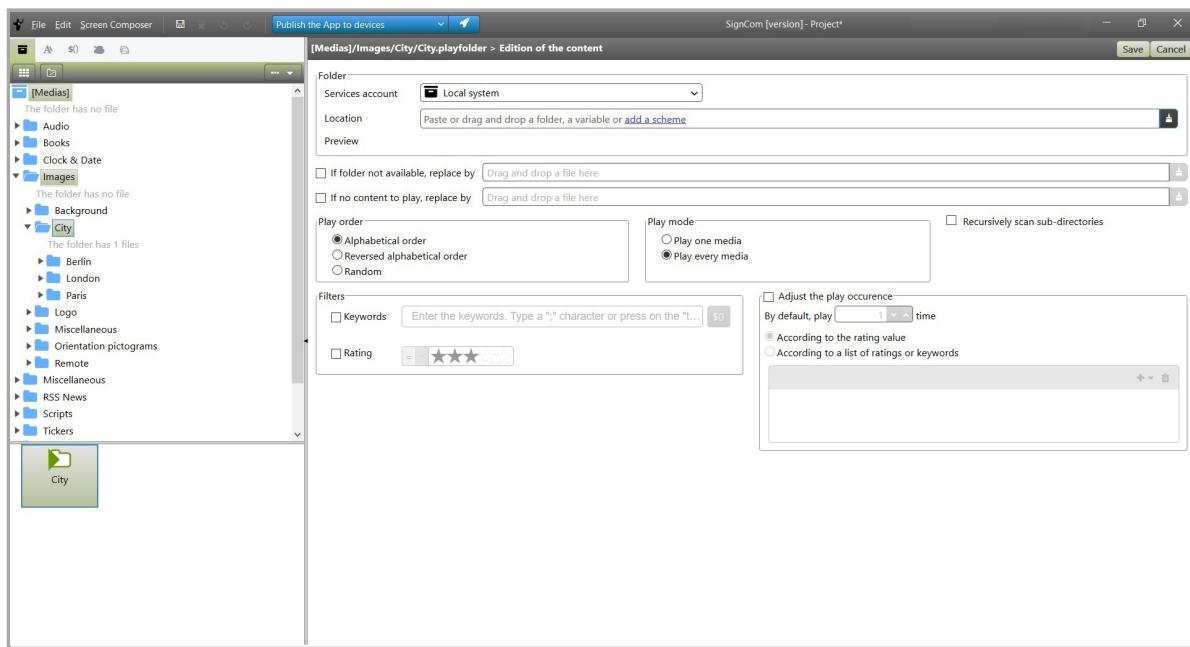
- Berlin,
- London,
- Paris.



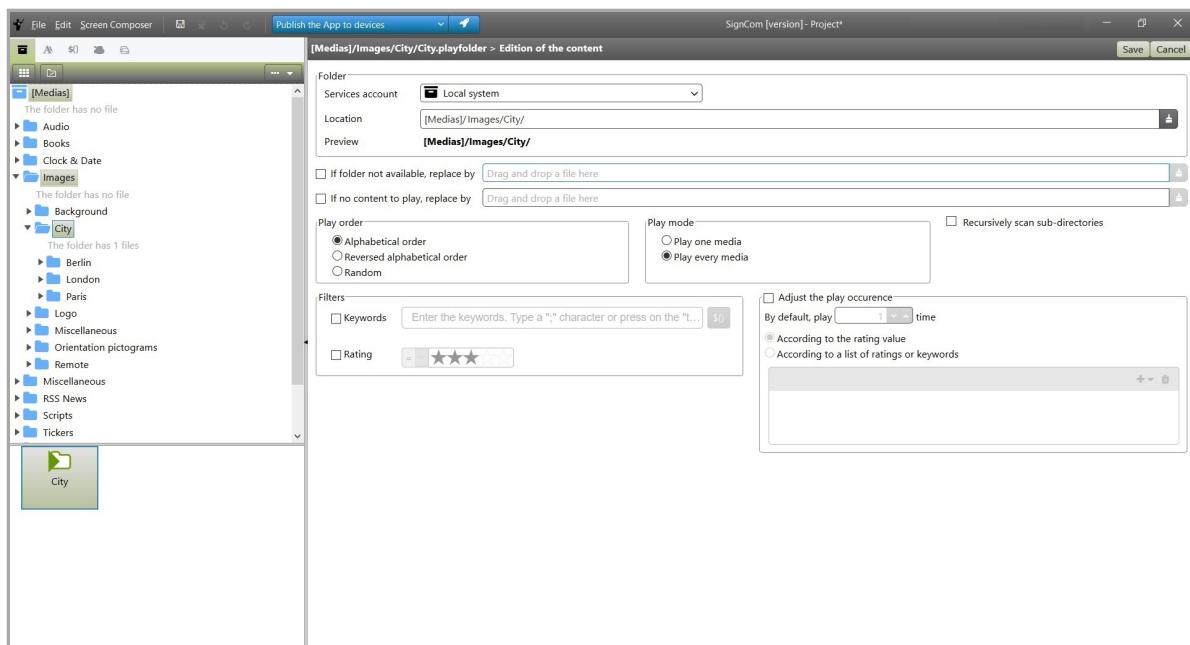
In the `Medias` tab, ensure to have a folder (e.g. *City* [1](#numstamp_1)) having subdirectories: - /Berlin, - /London, - /Paris.



To create the playfolder with variable, select an appropriate folder and right click on it to display its contextual menu. Select the Create a file item then click on the create a playfolder (1) item (e.g. /City).

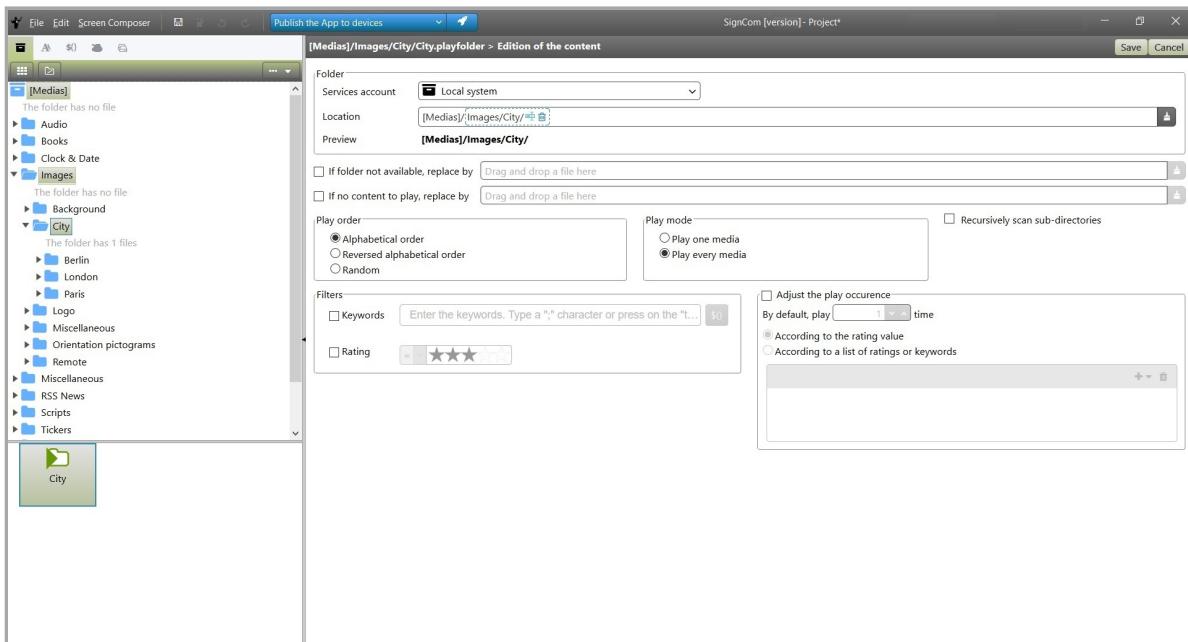


Drag and drop the folder located at the root of the 3 directories (5) (for example /City) from the Media tab to the location input of the playfolder (6).

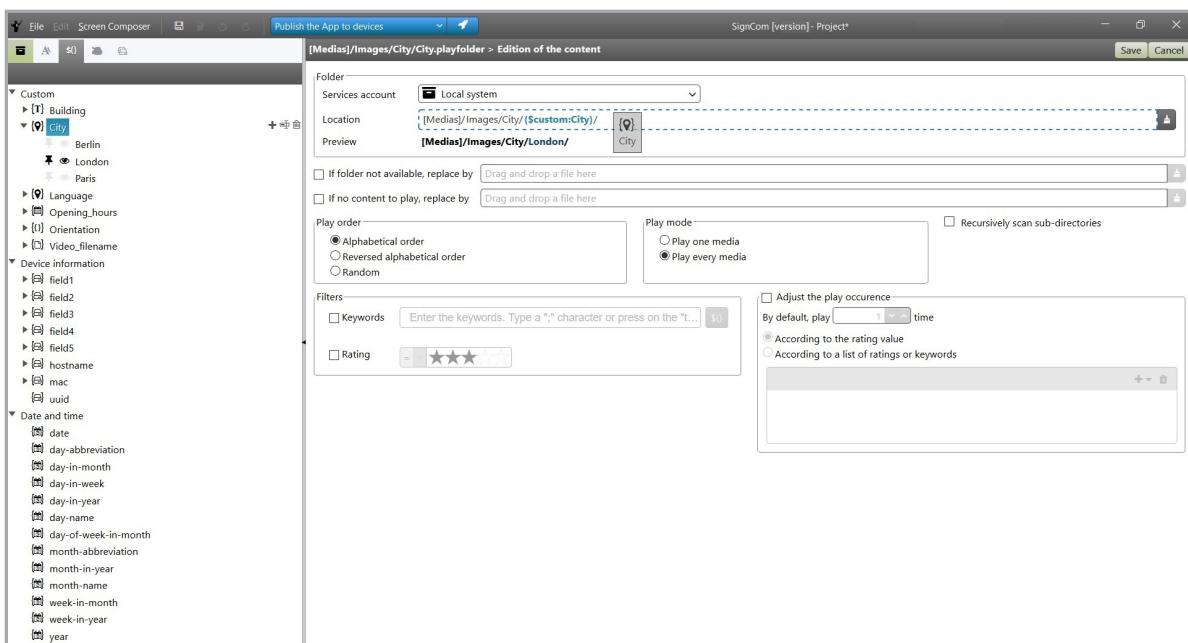


Select the **Variables** tab.

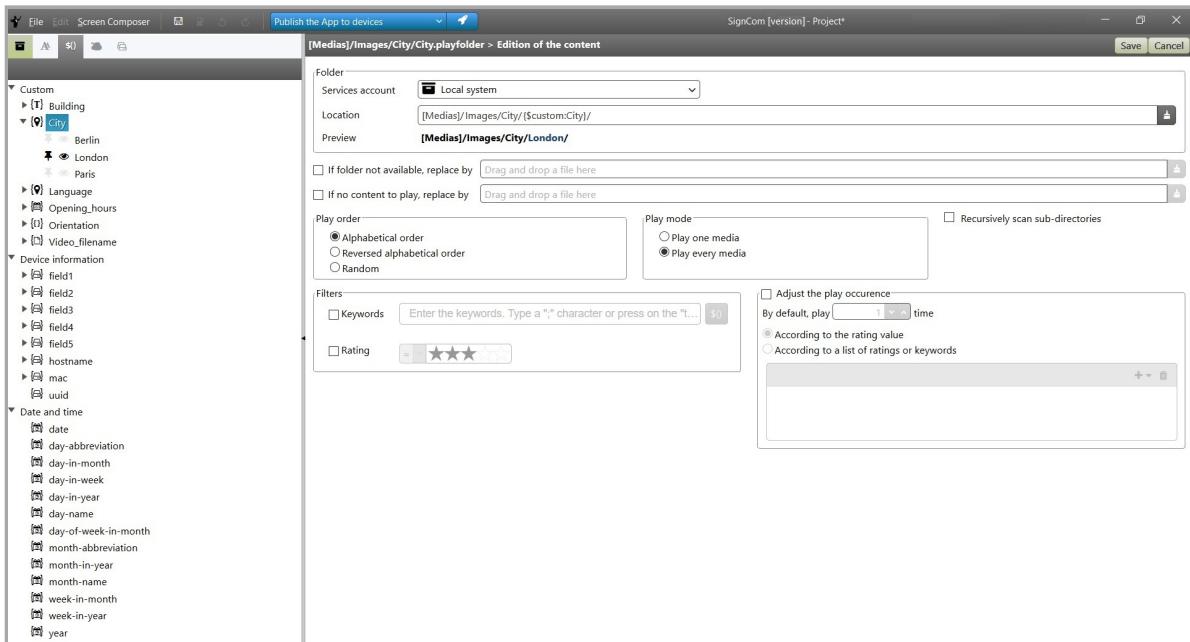
With your mouse, fly over the end of the playfolder location input to make appear the **Edit** blue button and clic on it.



Drop the wished **custom** localized variable name (e.g. **City**).

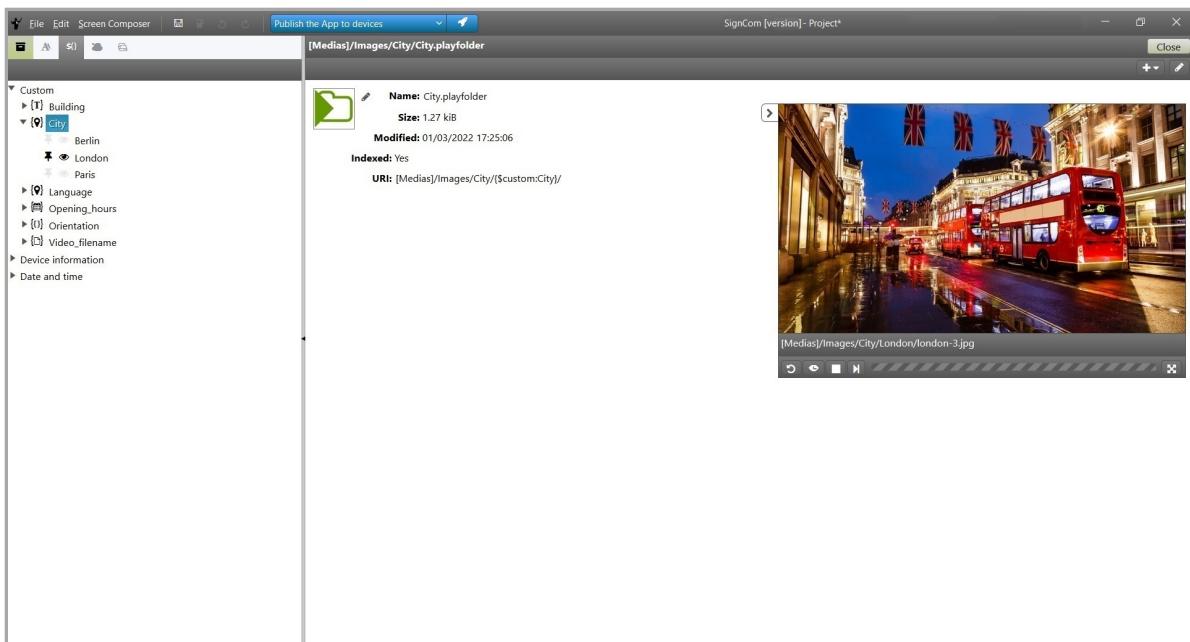


The preview of the playfolder is shown just below by using the preview variable value.



Save the playfolder (e.g. City). Launch the preview of the City playfolder. The content of the [Medias]/Images/City/London/ is played

Change the preview value of the City variable to preview another content.

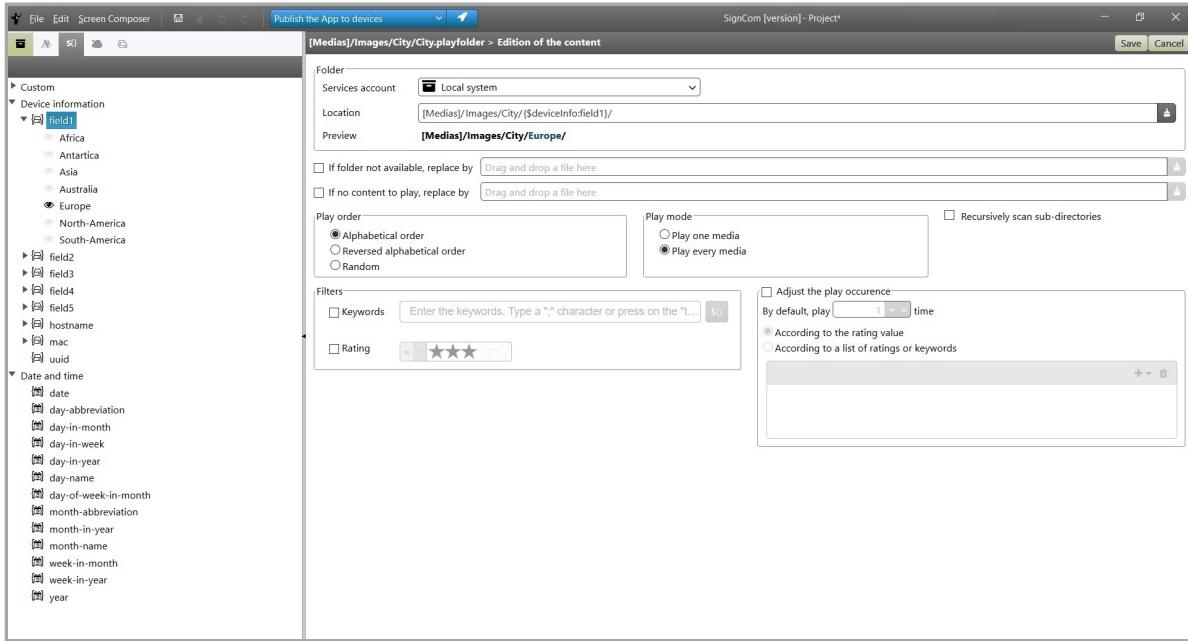


Depending on the custom variable `City` value published on the devices, the playfolder `City` will play the content of:

- `[Medias]/Images/City/Berlin/`,
- `[Medias]/Images/City/London/` or,
- `[Medias]/Images/City/Paris/`.

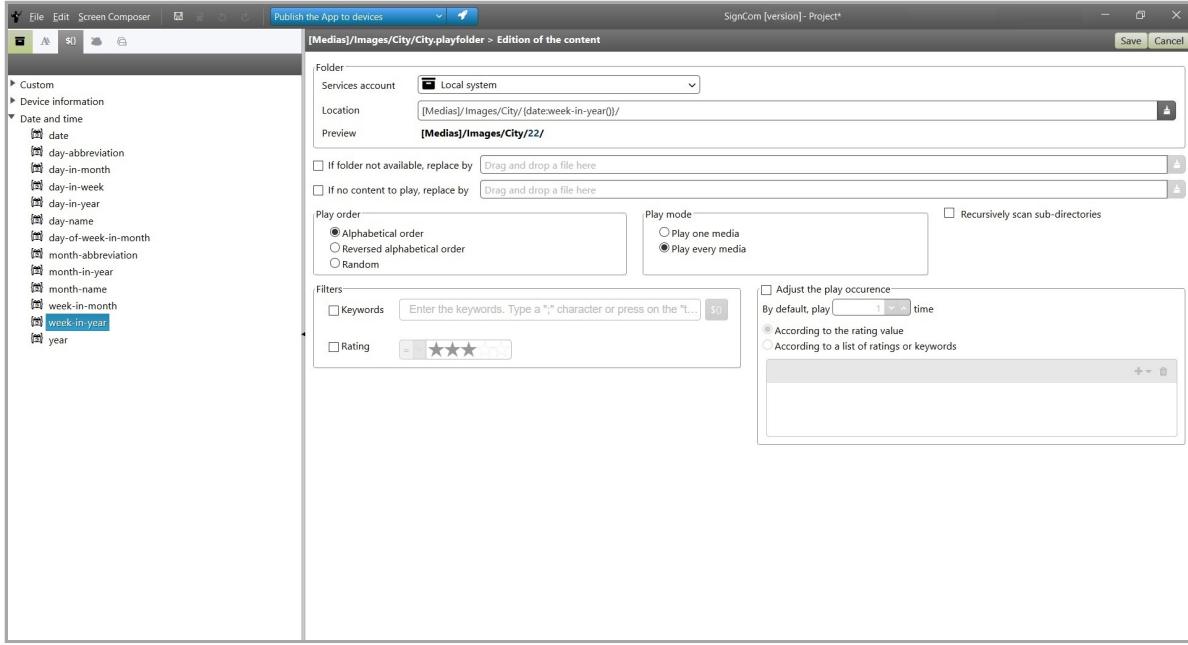
When using a `Custom` localized variable name, the `${custom:<variable_name>}` is pasted in the playfolder location input.

When using a `Device Information` variable name, the `${deviceInfo:<variable_name>}` is pasted in the playfolder location input (e.g. `${deviceInfo:field1}`).



⚠ Publishing Playfolder using a Device information variable name requires to add the manually the required target files or target folders in the manifest. For further information, refer to the chapter § [SignCom > Manifest](#).

When using a `Date and time` variable name, the `${date:<variable_name>}` is pasted in the playfolder location input (e.g. `${date:week-in-year()}`).



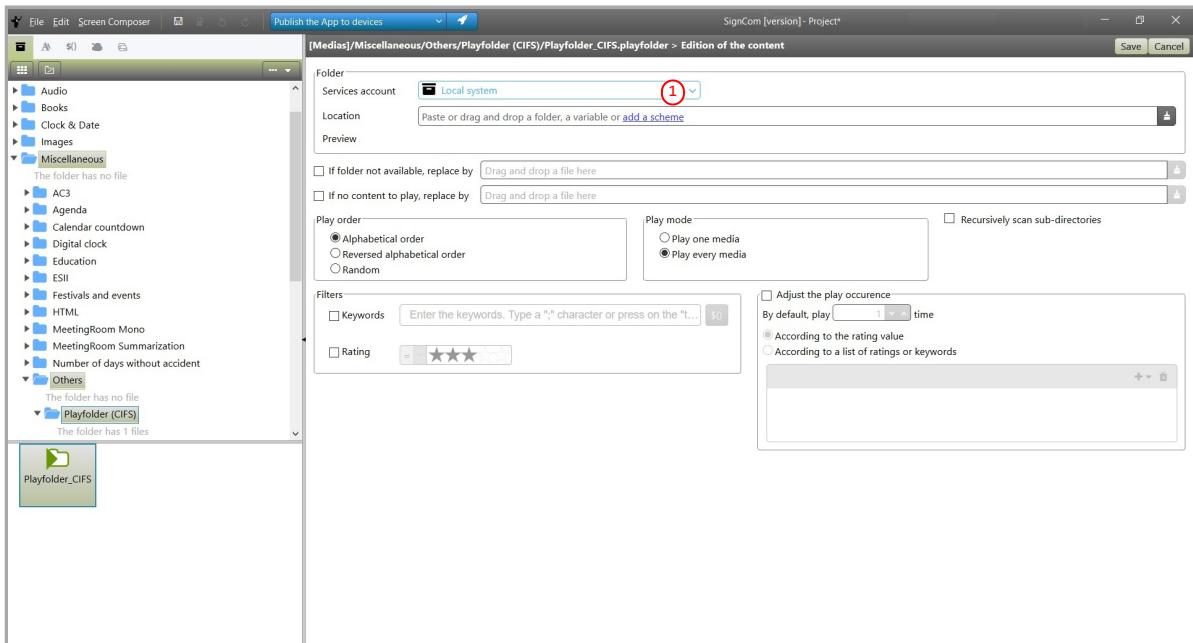
⚠ Publishing Playfolder using custom Date and time variable name requires to add the manually the required target files or target folders in the manifest. For further information, refer to the chapter § [SignCom > Manifest](#).

Play medias shared on your local network (CIFS)

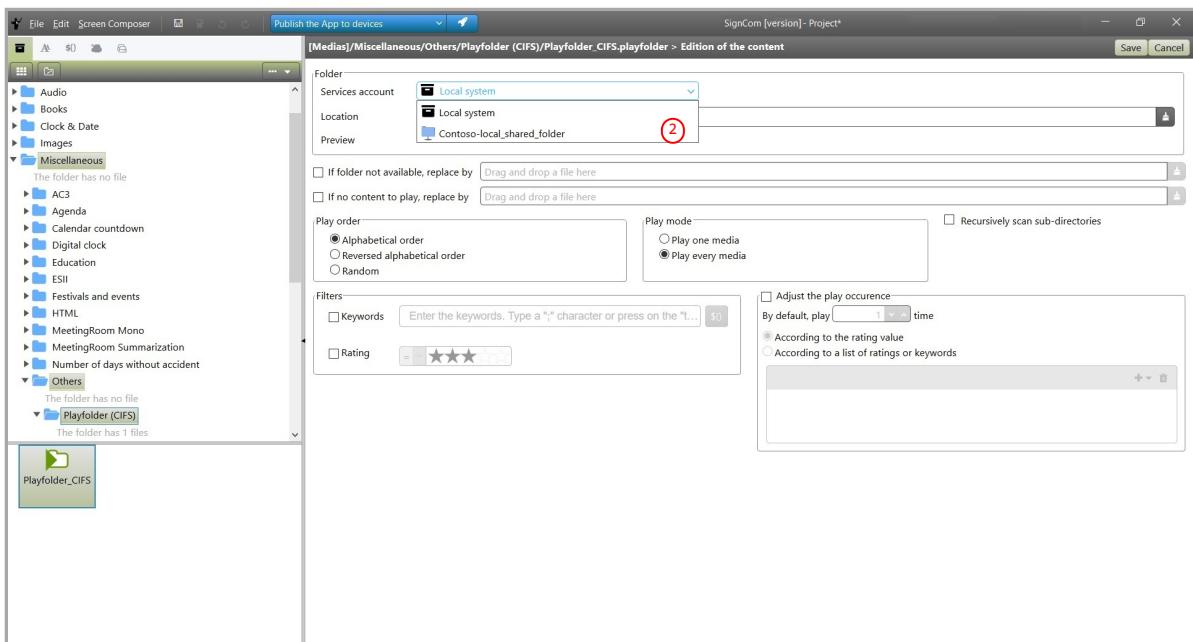
So that the `playfolder` can play medias from a shared directory available on your local network (*Microsoft SMB/CIFS*), a CIFS services account must be first created to handle authentication to access to the resources. For further information, refer to the chapter § [Services accounts](#).

Create a new `playfolder` (e.g. `Playfolder_CIFS`) as usual and select the `Services account` drop down list [**①**](#) in the `playfolder` form.

☞ The `Services account` value in the drop down list of any new `playfolder` is by default `Local system`, meaning that the `playfolder` plays by default medias which must be hosted in the repository.

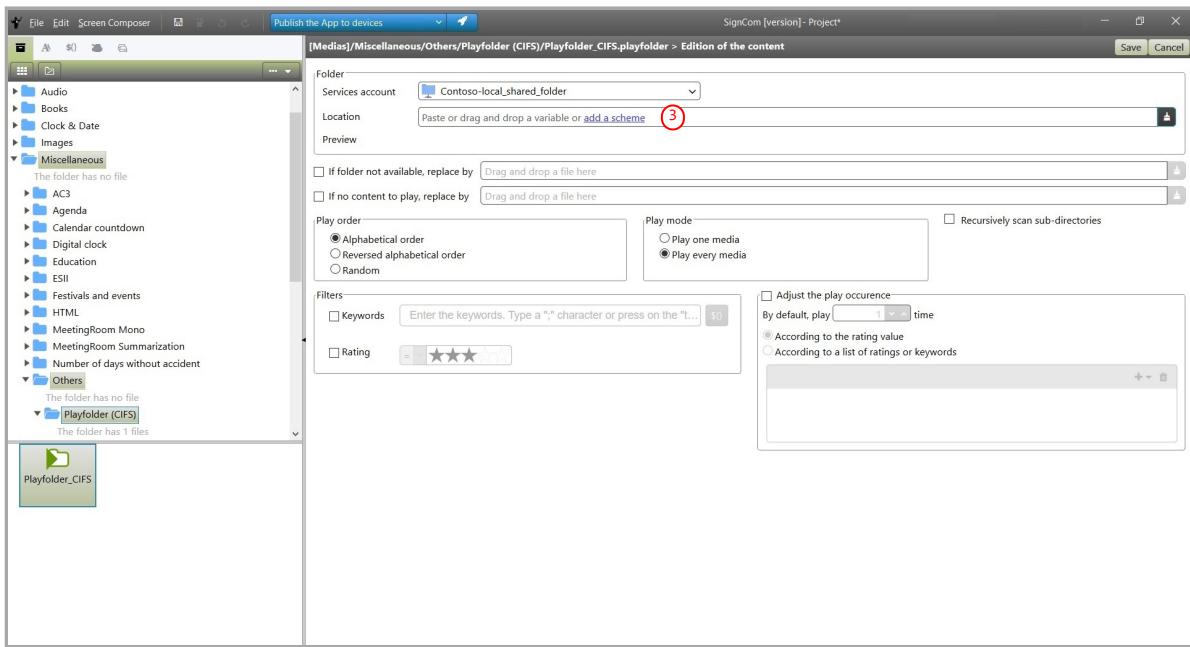


Select first one of the `CIFS` services accounts you've created (e.g. `Contoso-signcom_m365` [**②**](#)).

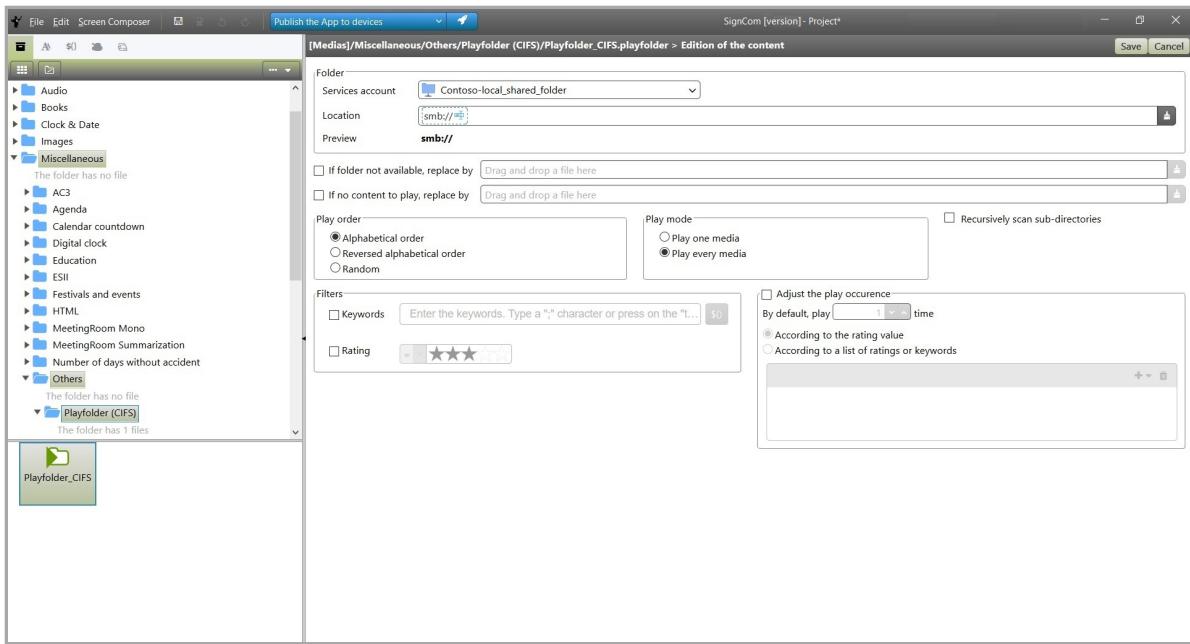


Click on the Add a scheme button.

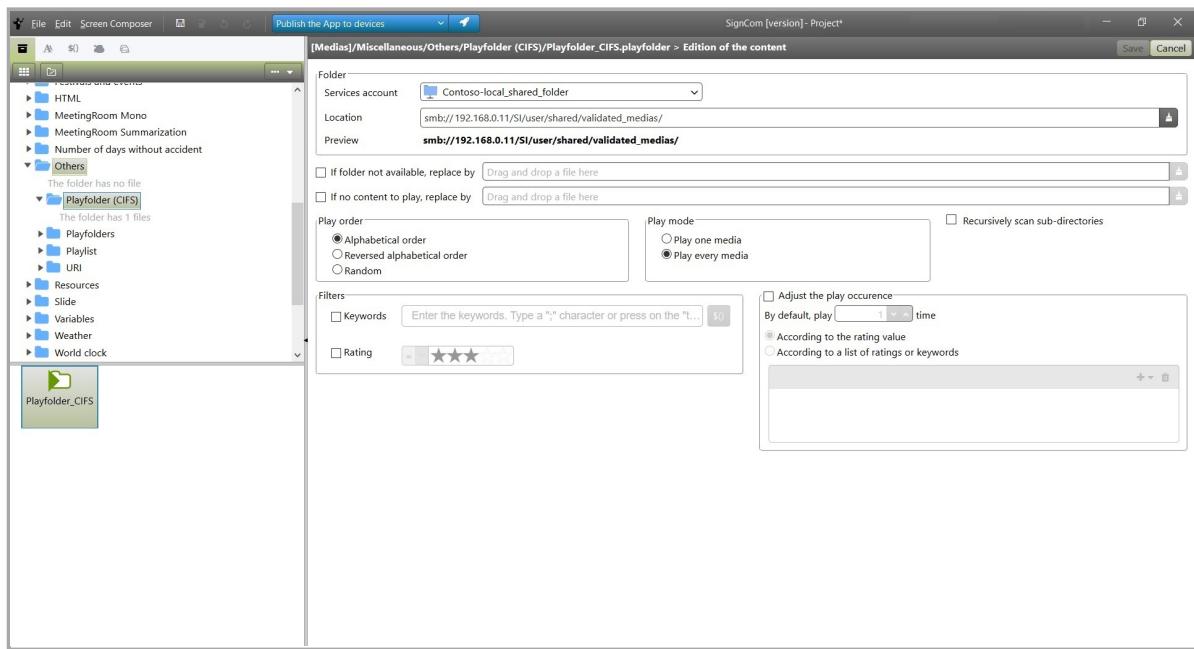
- When a CIFS services account is chosen, only the smb:// scheme is available.
- When associating another CIFS services account, the value of the Location input is cleared. If required, you can copy/paste the last Location input value by selecting the CIFS services account previously selected.



Click on the input, and fly over the smb:// scheme to make appear the blue edit button. Click on it to enter the path to access to your shared directory `smb://<machine_ip_addr>/<shared_directory>`.

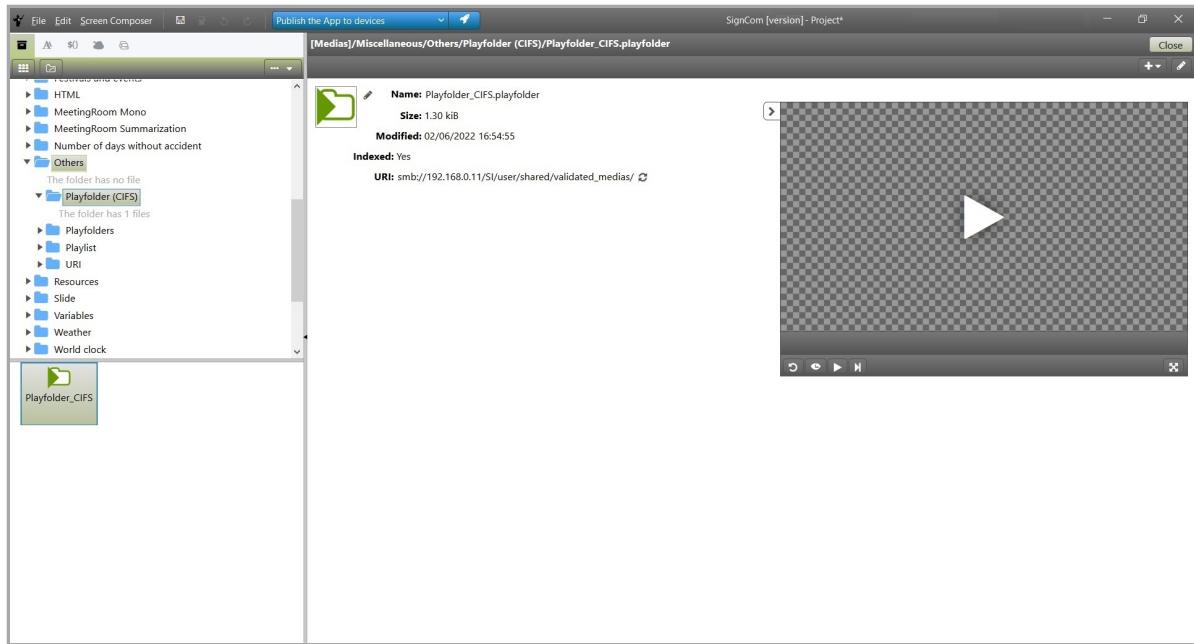


For example: smb://192.168.0.11/SI/user/shared/validated_medias/

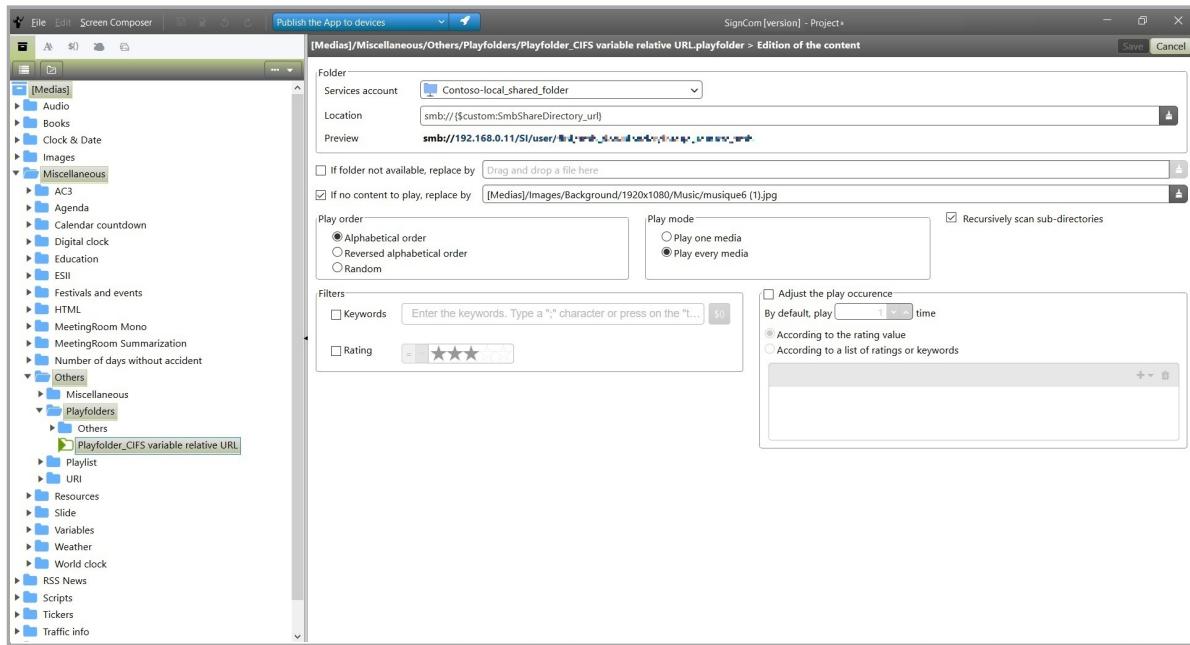


Once done, click on the Synchronize the folder with the services account for previewing button.

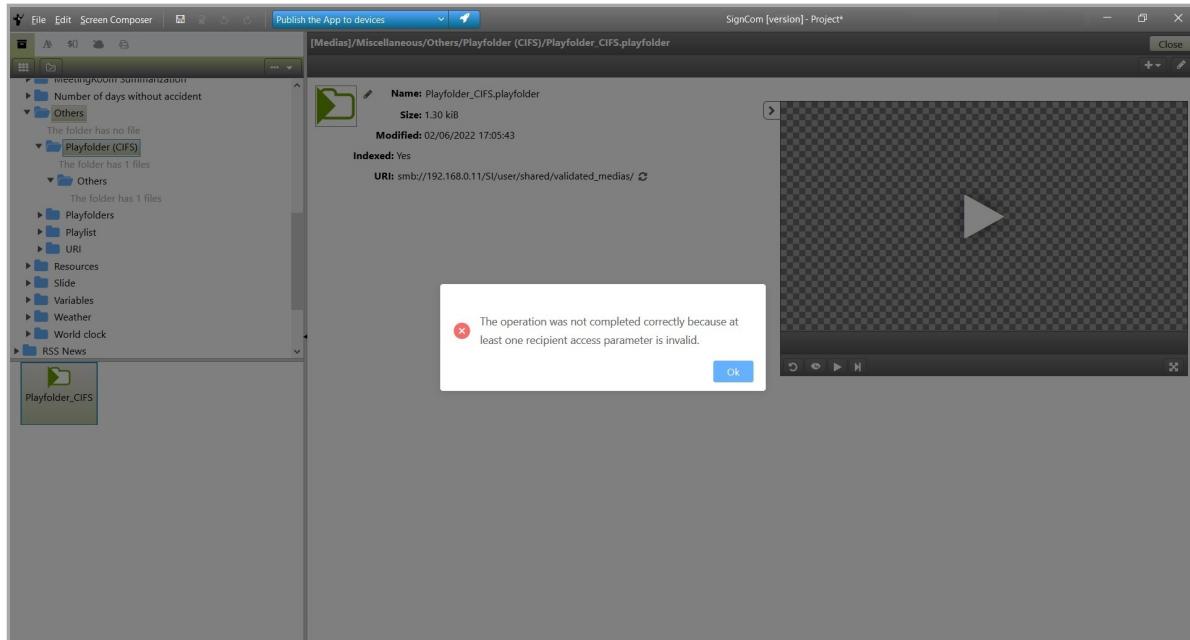
It is best to use the IP address of the machine sharing a directory rather than its netBios name because some device OS may not support netBios name resolution. For more information, refer to your device's user manual.



Example when using a `custom` variable of type `relative URL` :



If this error message appears at the time of synchronisation, check your shared directory access rights and services account credentials.



Once synchronisation is done, press on ► play button to preview the content of the playfolder.

⚠ When using a playfolder with Microsoft 365 services account or with services account CIFS, ensure that all media in the remote directory have an intrinsic duration defined and that the SignCom option Enforce on media with pagination a page progression with a default page duration is checked. If your playfolder hangs on playing media and is no longer able to play other media despite this, you must delete it from your remote directory and publish your App again.

⚠ If filtering by rating or filtering by variable is enabled for a playfolder with a Microsoft 365 services account or a CIFS services account, no media can be played. To work around this problem, disable filtering by rating and filtering by variable in your playfolder, and republish your application.

☞ Once published on the device, a periodic download task is automatically created. It is in charge of retrieving the remote content and storing it in a specific local directory `.medias\externals\<UUID_value>` dedicated to this playfolder .

☞ Supporting playfolder with CIFS services account requires to work with devices having an OS supporting the CIFS services account as well. For further information, contact support@innes.pro.

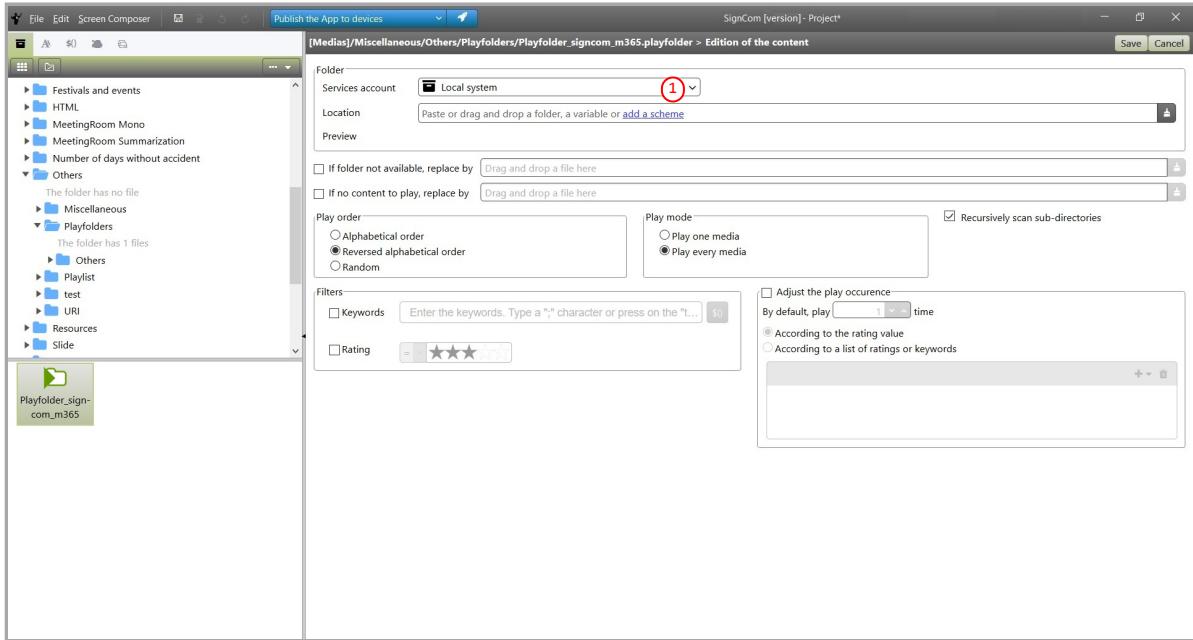
☞ CIFS services accounts are not supported on devices running with Gekkota OS 3.YY.ZZ.

Play medias located on your Microsoft 365/OneDrive

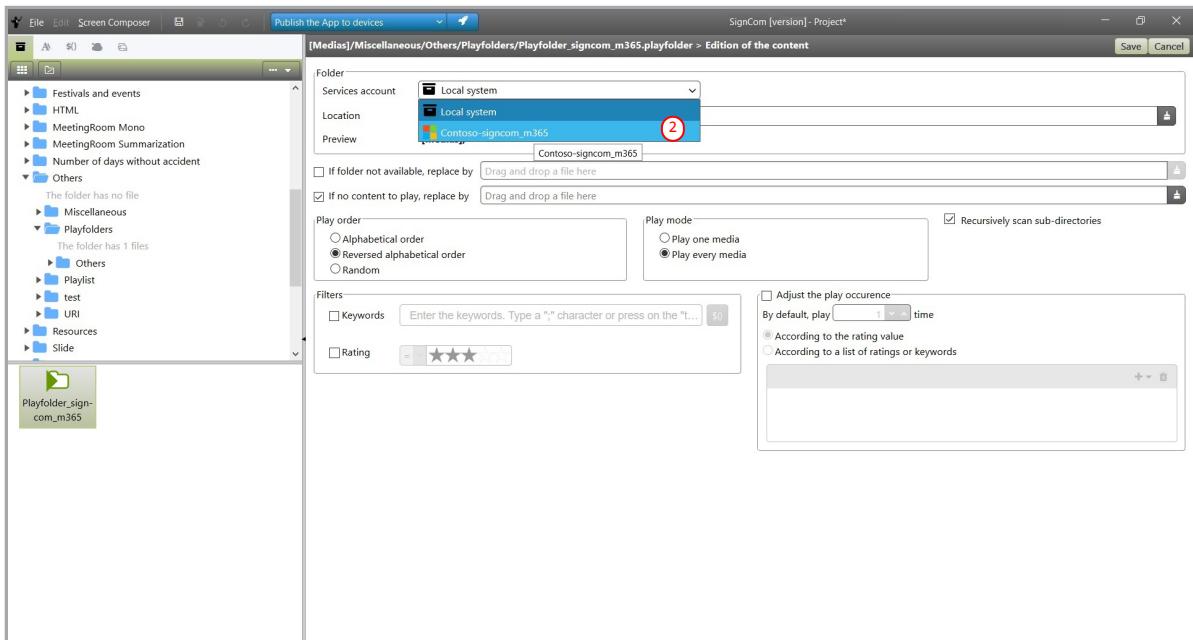
So that the playfolder can play medias located on a directory of your Microsoft 365 account, a *Microsoft 365 services account* must be first created to handle authentication to access to your Microsoft 365 resources. For further information, refer to the chapter § [Services accounts](#).

Create a new playfolder (e.g. *Playfolder-signcom_365*) and select the `services account` drop down list (1) in the playfolder form.

The services account value in the drop down list of any new playfolder is by default Local system, meaning that the playfolder plays by default medias that are hosted in the media repository.

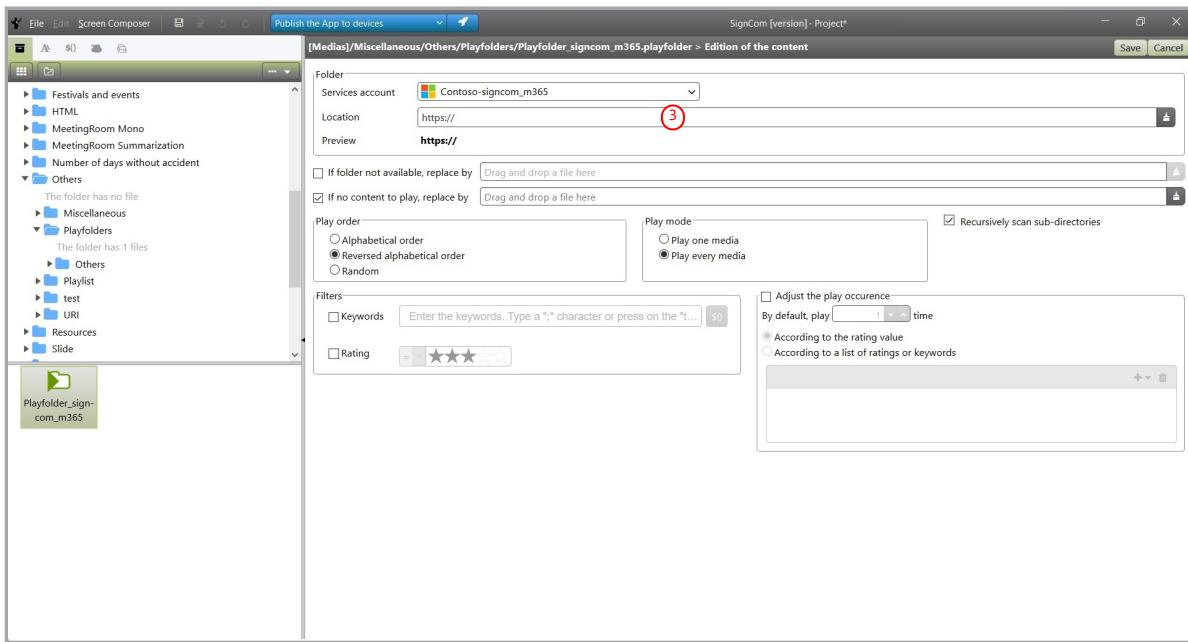


Select the Microsoft 365 services accounts you've created (e.g. *Contoso-signcom_m365* (2)).

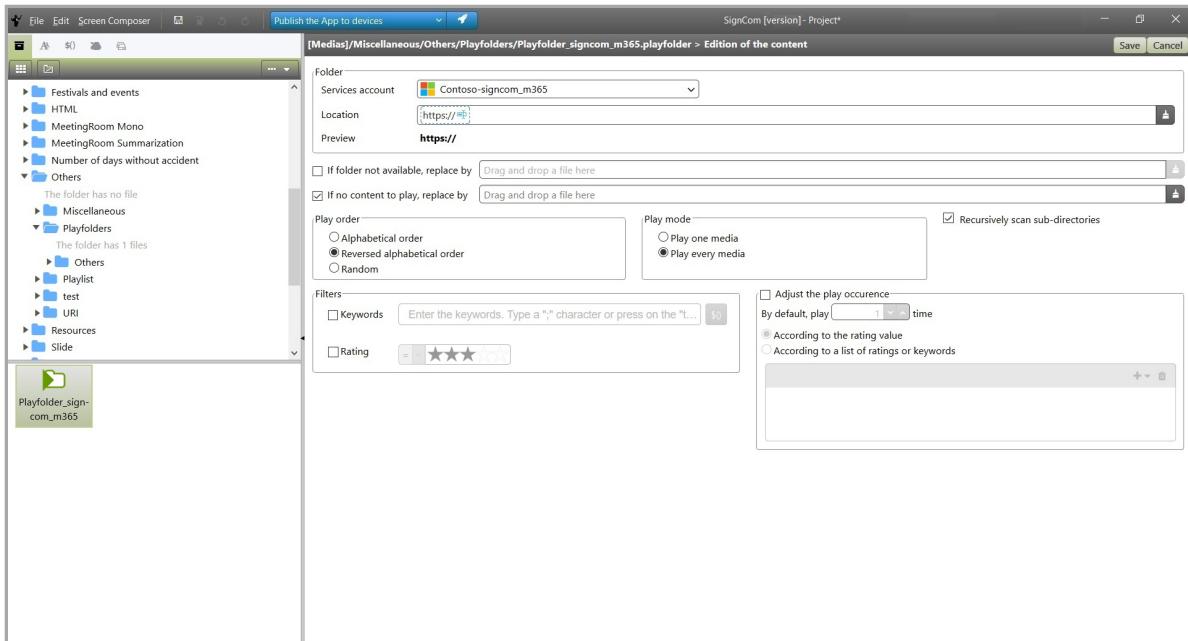


Click on the Add a scheme button.

- When a Microsoft 365 services account is selected, only the https:// scheme is available.
- When another Microsoft 365 services account is associated, the Location input value of the playfolder form is cleared. If necessary, you can copy/paste the last Location input value by selecting the previously selected Microsoft 365 services account .

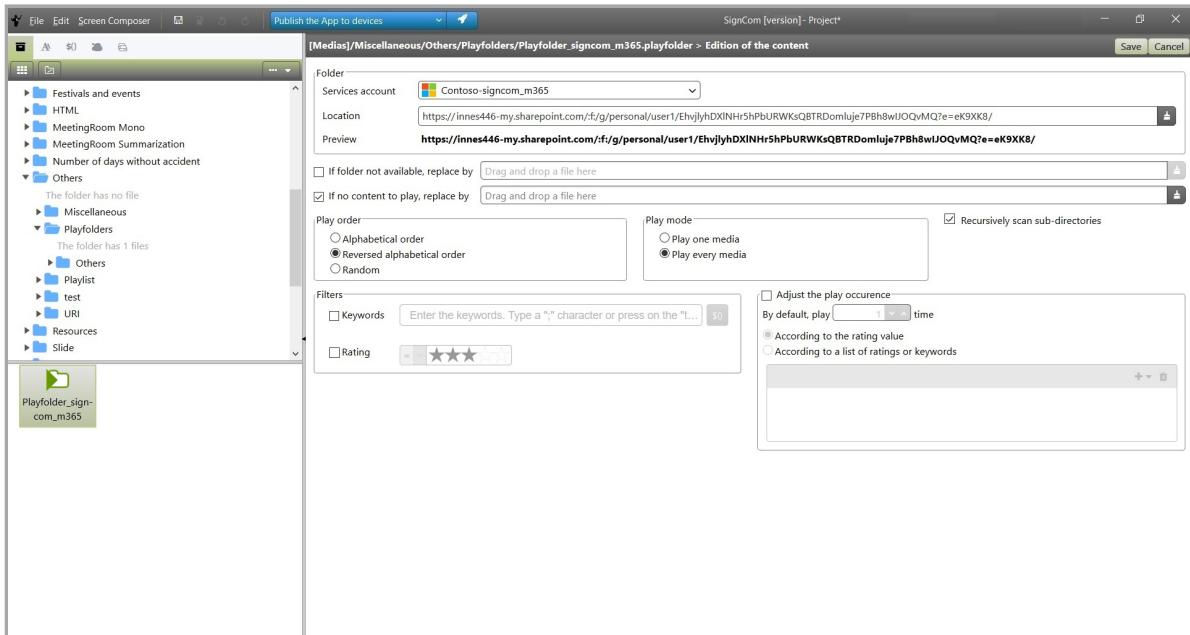


Clic on the location ③ input and paste, for example, a sharepoint link to your OneDrive folder available with the Microsoft 365 account used to created the Contoso-signcom_m365 services account .

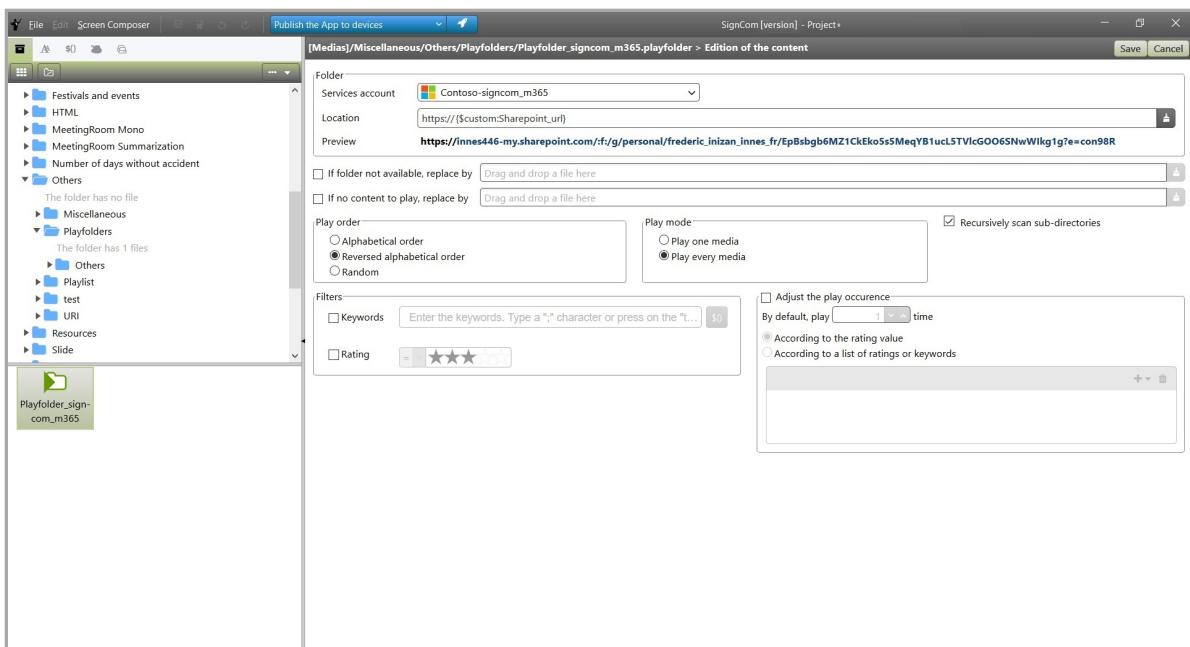


For example (fake): <https://innes446-my.sharepoint.com/:f/g/personal/user1/EhvjlyhDX1Nhr5hPbURWksQBTRDomluje7PBh8wIJOQvMQ?e=eK9Xk8>¹

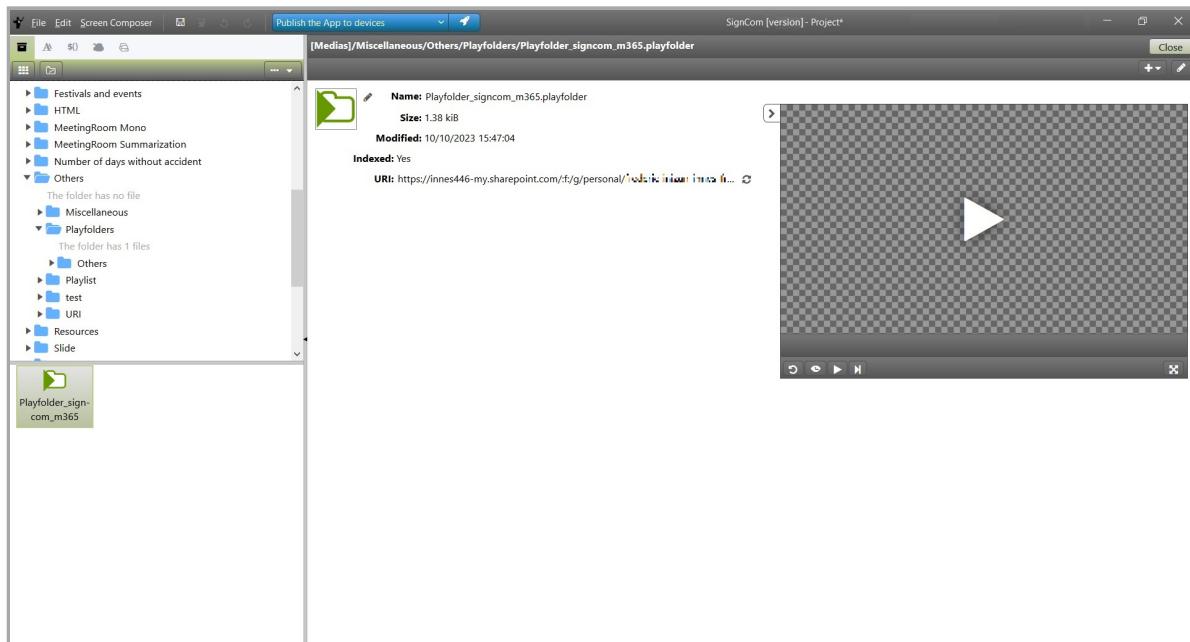
¹fake URL.



Example when using a `custom` variable of type `relative URL` :

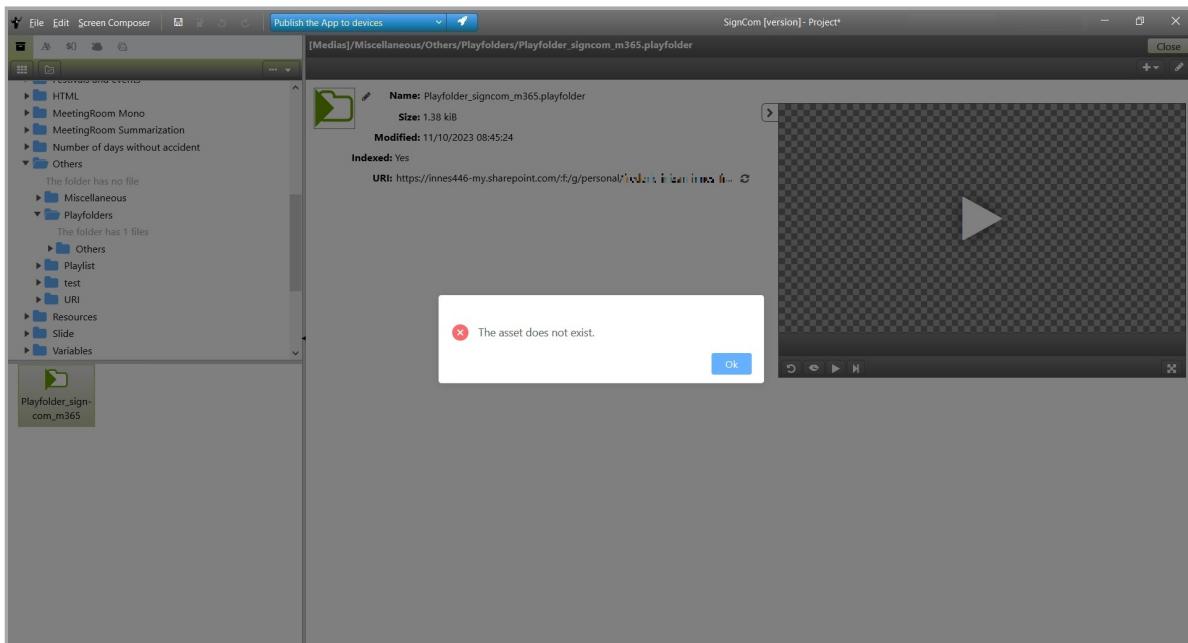


Once done, click on the Synchronize the folder with the services account for previewing button to download the content of the remote playfolder locally on your computer.



If this error message is prompted, check:

- the services account parameters values,
- the rights granted for your `signcom_m365` application.



Once synchronisation is complete, click on ► play button to preview the playfolder media.

⚠ When using a playfolder with a Microsoft 365 services account or with a CIFS services account, ensure that all media in the remote directory have an intrinsic duration defined and that the SignCom option Enforce on media with pagination a page progression with a default page duration is checked. If your playfolder hangs on playing media and is no longer able to play other media despite this, you must delete it from your remote directory and publish your App again.

⚠ If filtering by rating or filtering by variable is enabled for a playfolder with a Microsoft 365 services account or a CIFS services account, no media can be played. To work around this problem, disable filtering by rating and filtering by variable in your playfolder, and republish your application.

☞ Once published on the device, a periodic download task is automatically created. It is in charge of retrieving the remote content and storing it in a specific local directory `.medias\externals\<UUID_value>` dedicated to this playfolder.

☞ Supporting playfolder with Microsoft 365 services account requires to work with devices having an OS supporting the Microsoft 365 services accounts as well. For further information, contact support@innes.pro.

☞ Microsoft 365 services account s are not supported on devices running with Gekkota OS 3.YY.ZZ.

☞ The substitution media when the directory is not available anymore does not work for remote directory, for example when the remote directory has been removed and the link is broken. In this case, the playfolder continues to play the last content get during the last working folder synchronization. If the content of your playfolder seems not be synchronized anymore, double check the folder synchronization of your playfolder in the unitary preview window. If the message the asset does not exist is prompted, that means that the link is not consistent anymore.

☞ When your SignCom App is using playfolders with Microsoft 365 services accounts , it is mandatory to update first all your Gekkota device fleet with a version 4.14.10 (or higher) else the SignCom App may be not able to start after publication.

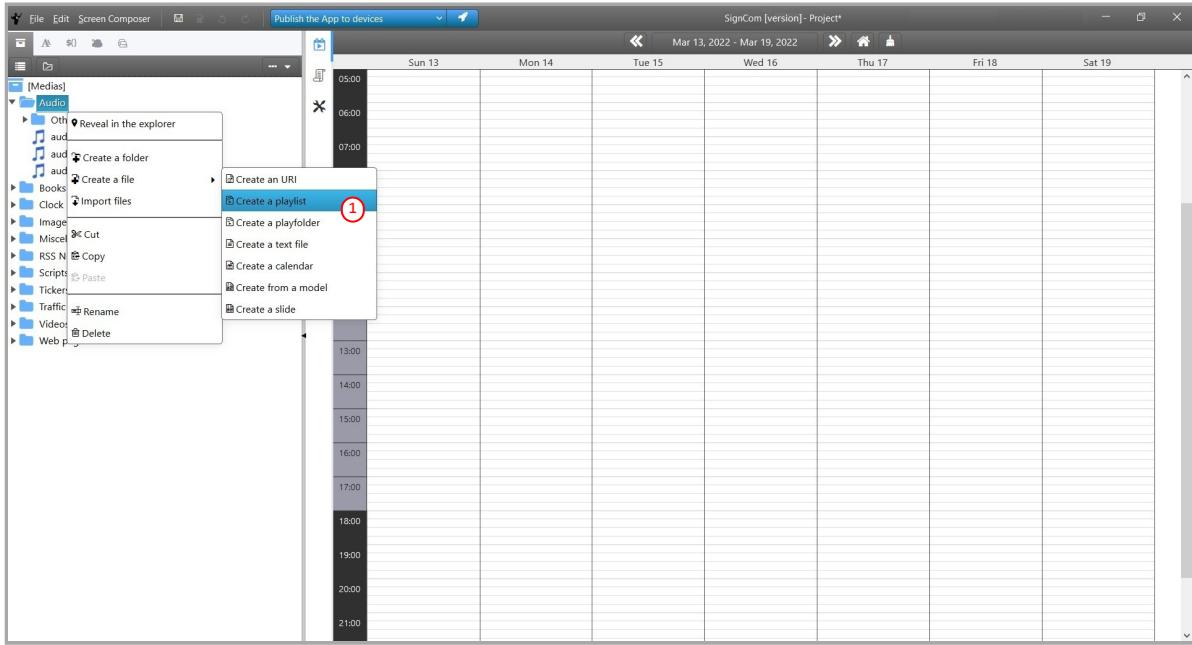
3.2.4 Medias > Playlist

A playlist media has a `.playlist` file extension. It allows to create a media which is playing some medias inserted in a list defined by the user by following some playing rules.

The playlist media supports for audio medias and visual medias. It is advised to create playlist medias containing either only audio medias, or either only visual medias.

Importing and maintaining playlists in `.xspf` format is supported, but creating playlist in the `.xspf` format is not supported. The XSPF format is documented on www.xspf.org. The root of the folder where the media objects to be played are located is described using an `xml:base` (<http://www.w3.org/TR/xmlbase/>).

To create a `.playlist` media, select an appropriate folder and right click on it to display its contextual menu. Select the  Create a file item then click on the  Create a playlist **①** item.



Enter a name for the `.playlist` file (e.g. `Audio`) then click on the `ok` button.

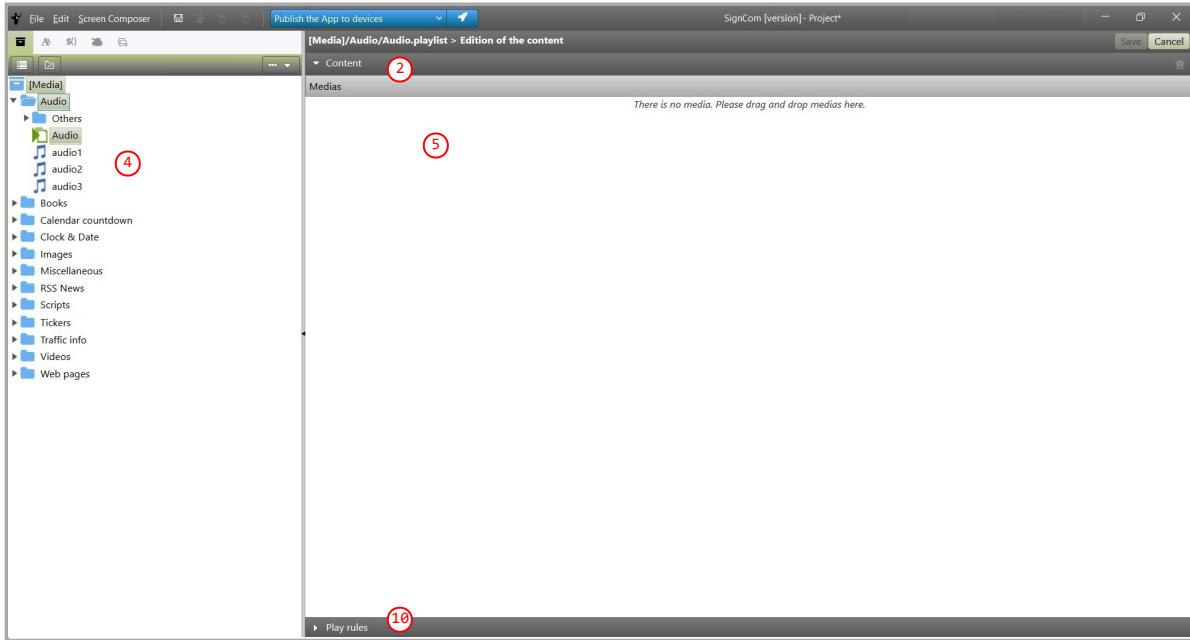
Content

The playlist form consists in two panes:

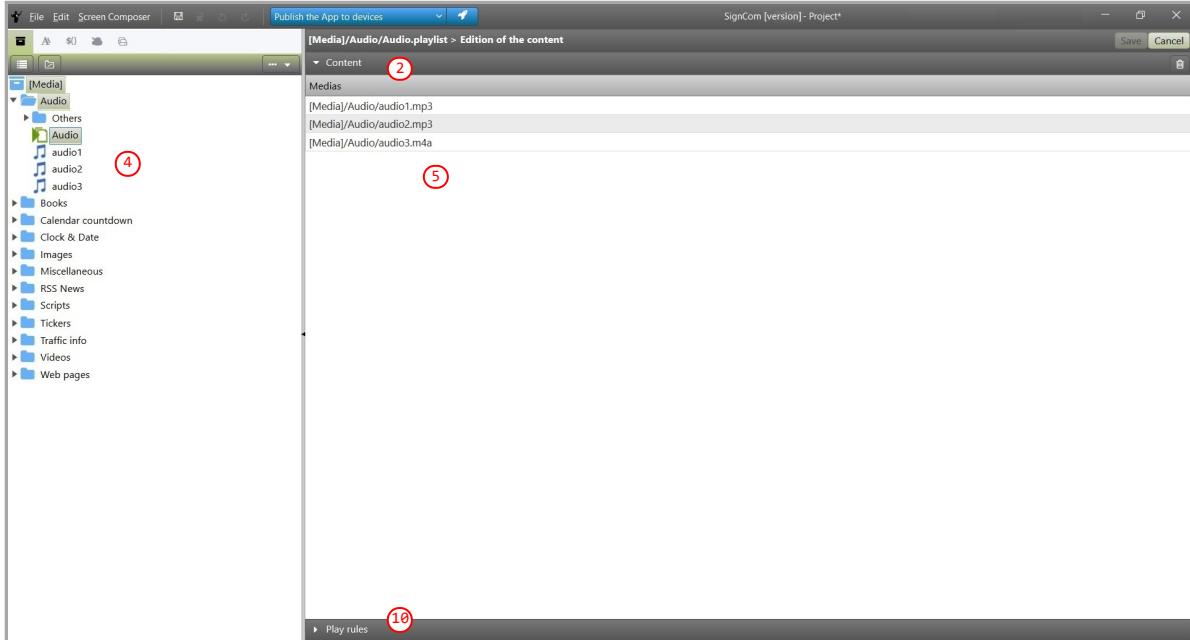
- Content (2)
- Play rules (10).

☞ Click on the Content (2) or Play rules (10) pane banner to collapse or expand the pane.

To add the list of the medias to play in the playlist media, drag'n drop medias from the Media tab (4) to the Content pane (5) on the right.



☞ It is not possible to drag'n drop .uri medias, .playlist medias and .playfolder medias inside the playlist medias list.



Playing rules

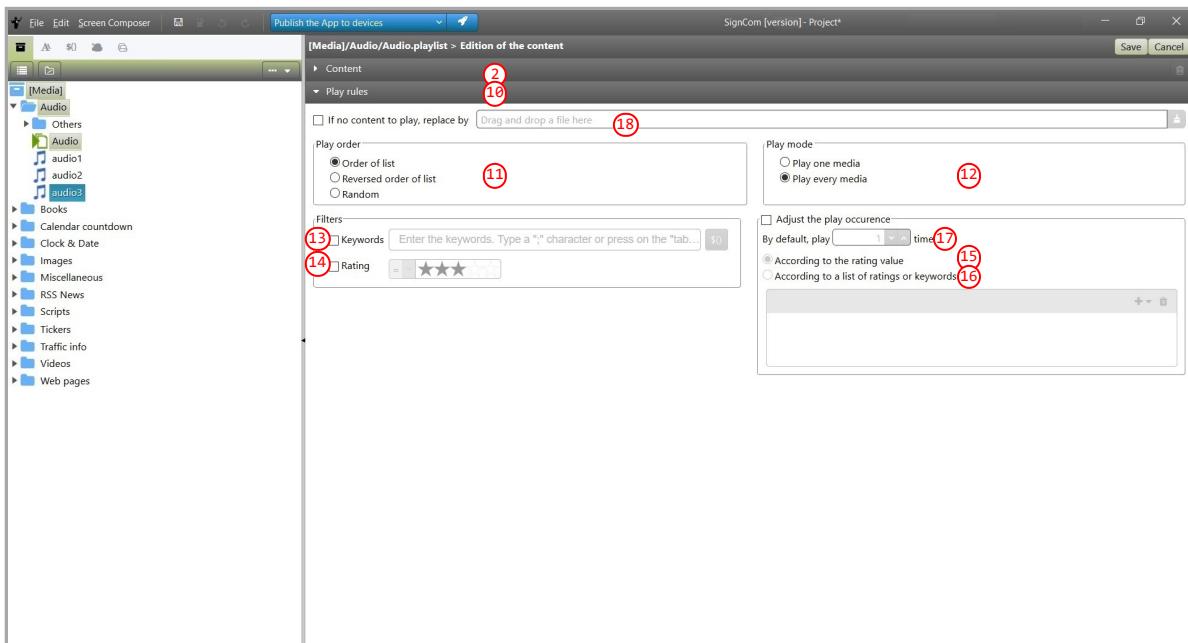
The playlist form consists in two panes:

- Content **(2)**,
- Play rules **(10)**.

☞ Click on the Content **(2)** or Play rules **(10)** pane banner to collapse or expand the pane.

Click on the Play rules **(10)** pane banner to expand the Play rules pane.

- Play order **(11)**:
 - Play the medias in
 - Order of list
 - Reversed order of list,
 - Random,
- Play mode **(12)**:
 - Play one media,
 - Play every media,
- Filters : allows to display only some medias based on some rating or keyword values conditions:
 - Keyword **(13)**: allows to play only the medias that have the specified keyword or the set of keywords.
 - Rating **(14)**: allows to play only the medias that have some metadata rating values.
 - equal (=), lower (<) or higher (>) than a rating value
 - possible rating values: from 1 to 5 stars
- Adjusting the play occurrence :
 - According to the rating value **(15)** allows to play:
 - Five times in a row, the medias with a 5-stars rating,
 - Four times in a row, the medias with a 4-stars rating,
 - Three times in a row, the medias with a 3-stars rating,
 - Two times in a row, the medias with a 2-stars rating,
 - Once, the medias with a 1-star rating,
 - Following a list of ratings or keywords ³ **(16)** allows to play:
 - <n1> times in a row, the medias with a 5-stars rating,
 - <n2> times in a row, the medias with a 4-stars rating,
 - <n3> times in a row, the medias with a 3-stars rating,
 - <n4> times in a row, the medias with a 2-stars rating,
 - <n5> times, the medias with a keyword value 1,
 - <n6> times, the medias with a keyword value 2 and with a rating value 2, ...
 - By default, play <n> times **(17)**: allows to play <n> times in a row, the medias with no special rules.
- Case at the limits:
 - **(18)** Play an alternative media when the playlist is empty.



☞ For playlist medias playing visual content, the playlist background is defined in its behaviour property when the playlist is inserted inside a calendar time slot.

² When both keyword and rating filters are enabled, the media to be played must meet both conditions. When multiple keywords are specified in the filter, all keywords values must be present in a media for it to be played.

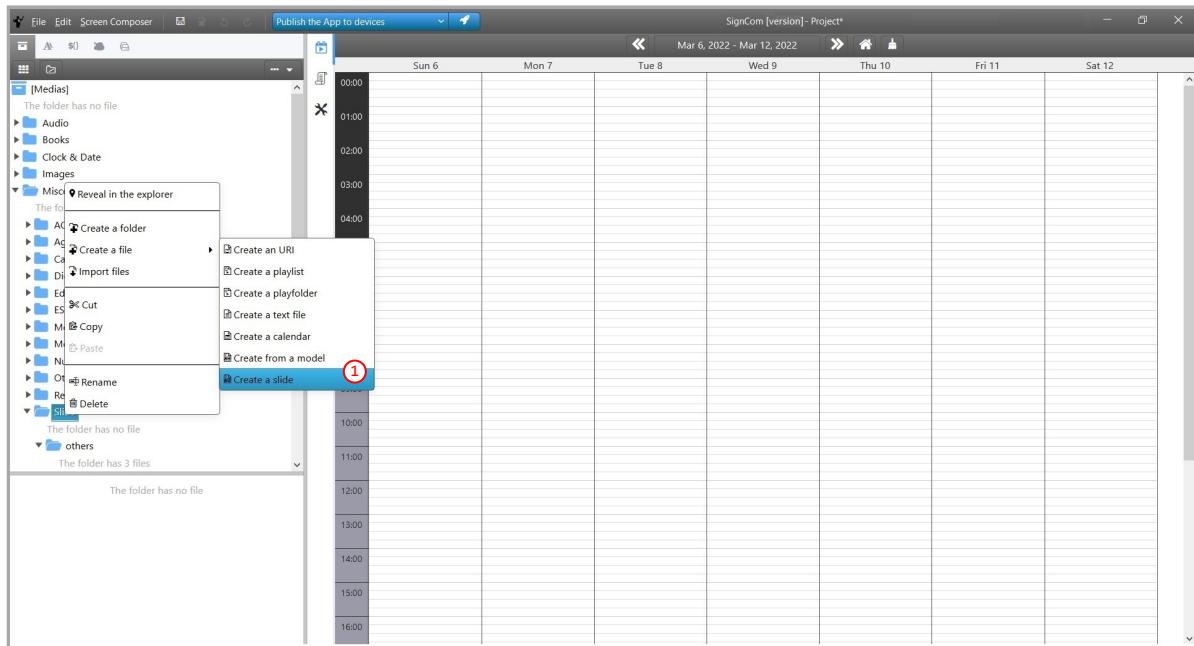
Limitations

It is not possible to drop a .playlist media, a .playfolder media or a URI media inside a .playlist form input.

3.2.5 Medias > Slide

The **Slide** tool allows to create simply a slide show which is displaying a content changing along the time by following a timeline.

To create a **slide** media (.wgt), select an appropriate folder and right click on it to display its contextual menu. Select the **Create a file** item then click on the **Create a slide** (1) item.



Enter a name for the **.wgt** file (e.g. *Slide*) then click on the **ok** button.

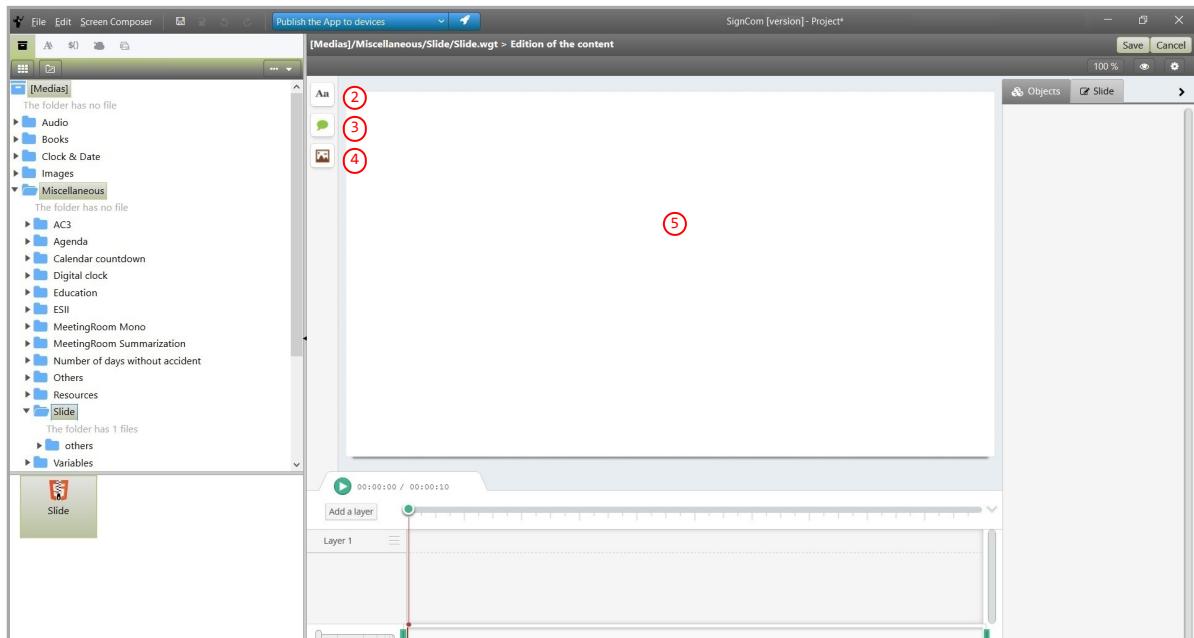
The objects supported in the **Slide** tool are:

- images,
- popups (with about twenty bubble models),
- text boxes.

To create a **text**, drag and drop the **Aa** (2) button to the slide host area (5).

To create a **popup**, drag and drop the bubble button (3) to the slide host area (5).

To create a **image** host, drag and drop the image button (4) to the slide host area (5).

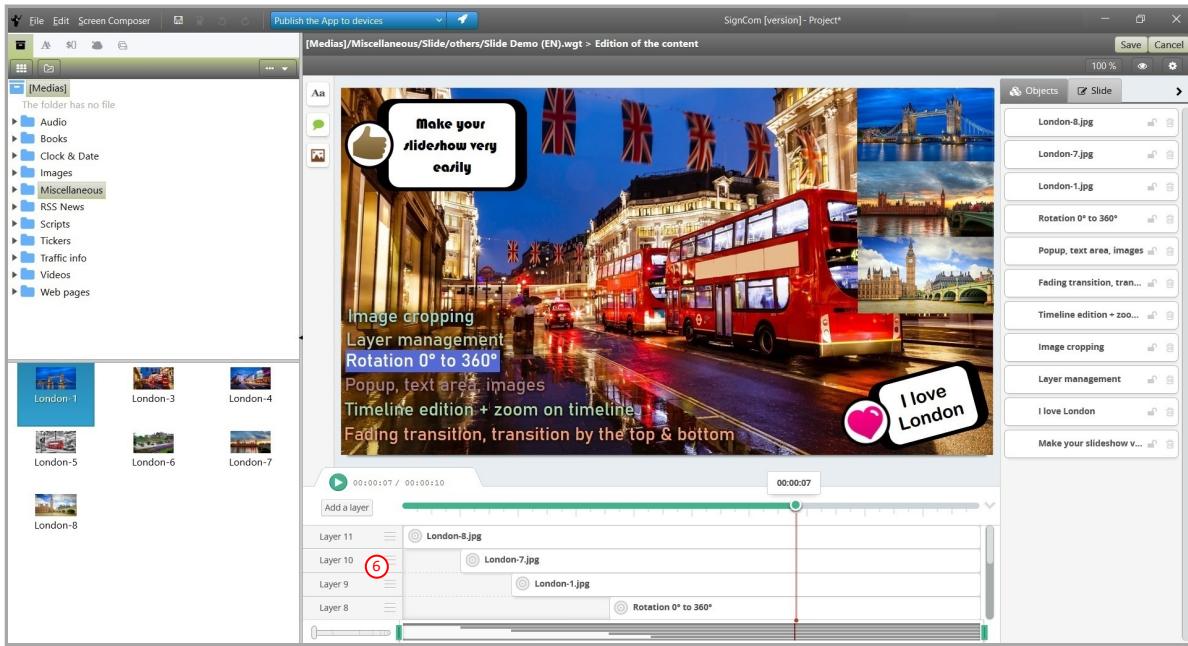


The supported functions include:

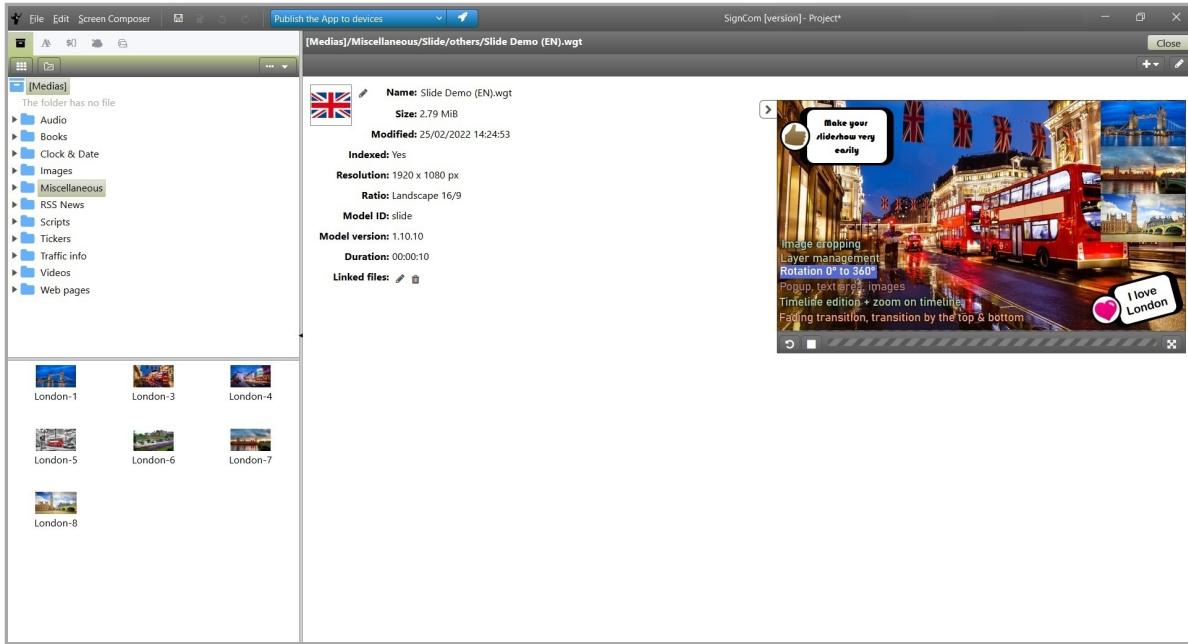
- the fade in, fade out, slide up or slide down transition,
- the rotation of objects (to the nearest degree),
- the image cropping,
- the possibility to change the style (fonts, size, shadow, color).

The duration of the slideshow is programmable. It is possible to zoom in on an area of the timeline to make finer time or start adjustments. The play / pause button allows to launch a preview inside the editor without saving. The layer manager allows to manage the depth of the different layers.

Each time a new object is created, a layer (6) is created.



Preview the different contents of your slide show along the time line.

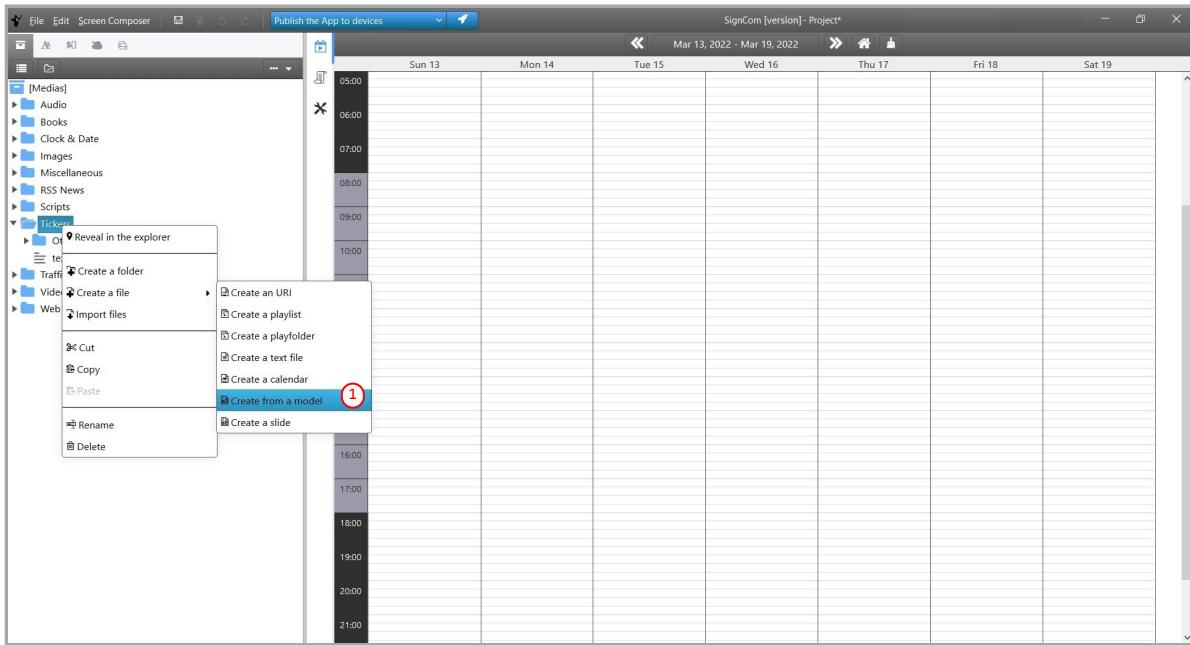


3.2.6 Medias > Create from a model

With the **Create a model** button, it is possible to:

- create **HTML widgets** from content models,
- create **URI** from URI models.

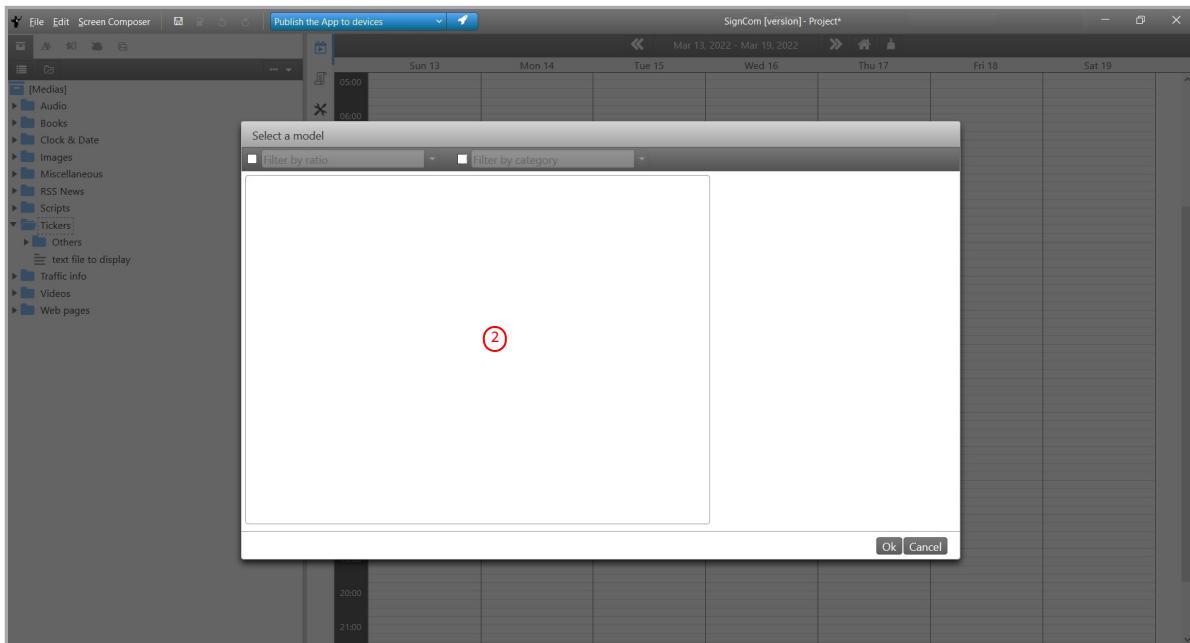
To create an HTML widget based on a content model, select a folder and right click on it to display its contextual menu. Select the **Create a file** item then click on the **Create from a model** (1) item.



By default, the library of content models in INNES Screen Composer G4 is empty (1). Depending on the required HTML widget you need to create, you may install only one or all of the available content model packs among the *Education*, *Traffic Info France*, *General* and *Signage* pack categories. For further information, refer to the chapter § [Screen Composer > Preferences > Repositories](#).

By default, the library of URI models in INNES Screen Composer G4 is empty (2). Depending on the required URI you need to create, you may install only one or all of the available URI model packs among the *Web TV*, *TV channels- France (DVB-T demodulator)*, *Device peripherals* pack categories. For further information, refer to the chapter § [Screen Composer > Preferences > Repositories](#).

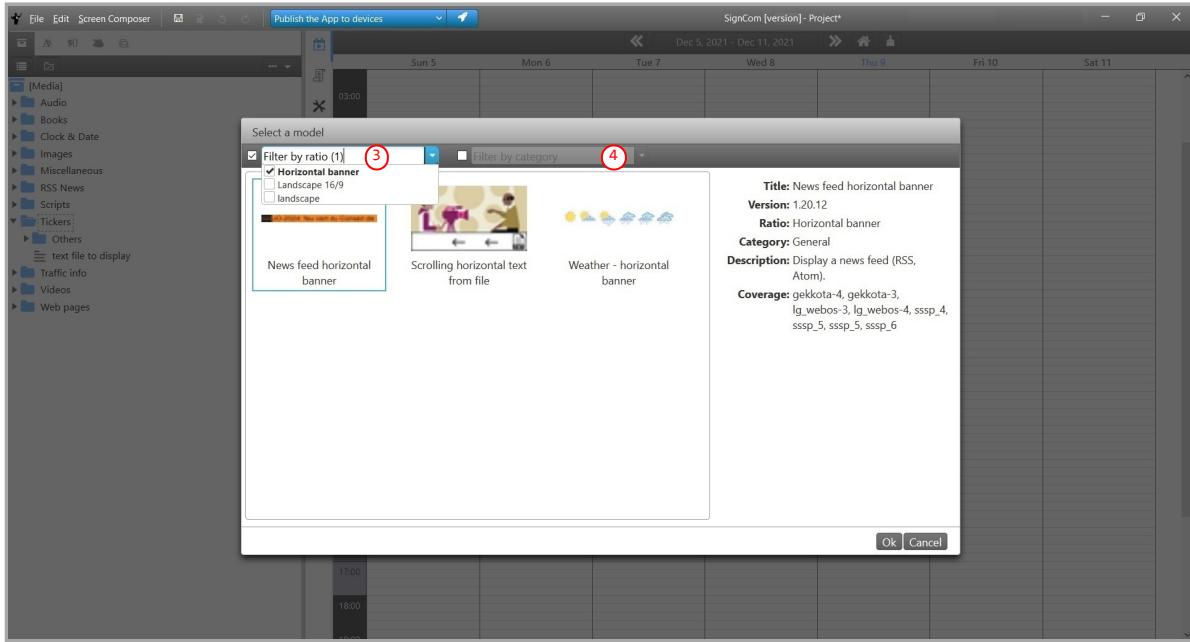
For further information about URI creation from URI models, refer to the chapter § [Medias > URI](#).



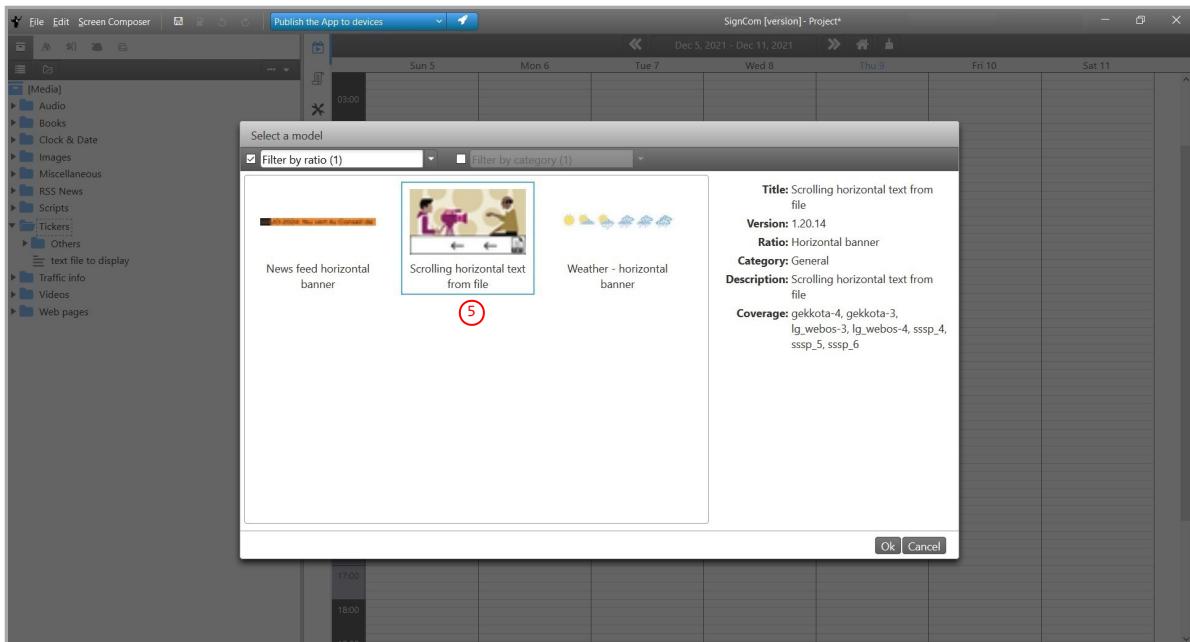
After having been installed, the content models are available in the content model library. Afterwards it is possible to create HTML widgets from them.

A short description of the content model is described in the chapter § [Appendix: Content model short description](#).

When some models are available, it is possible with the category (3) or ratio (4) filters to display only a subset of the content models. By default, no filter is activated.



Select then the wished content model (e.g. *Scrolling horizontal text from file* (5)).

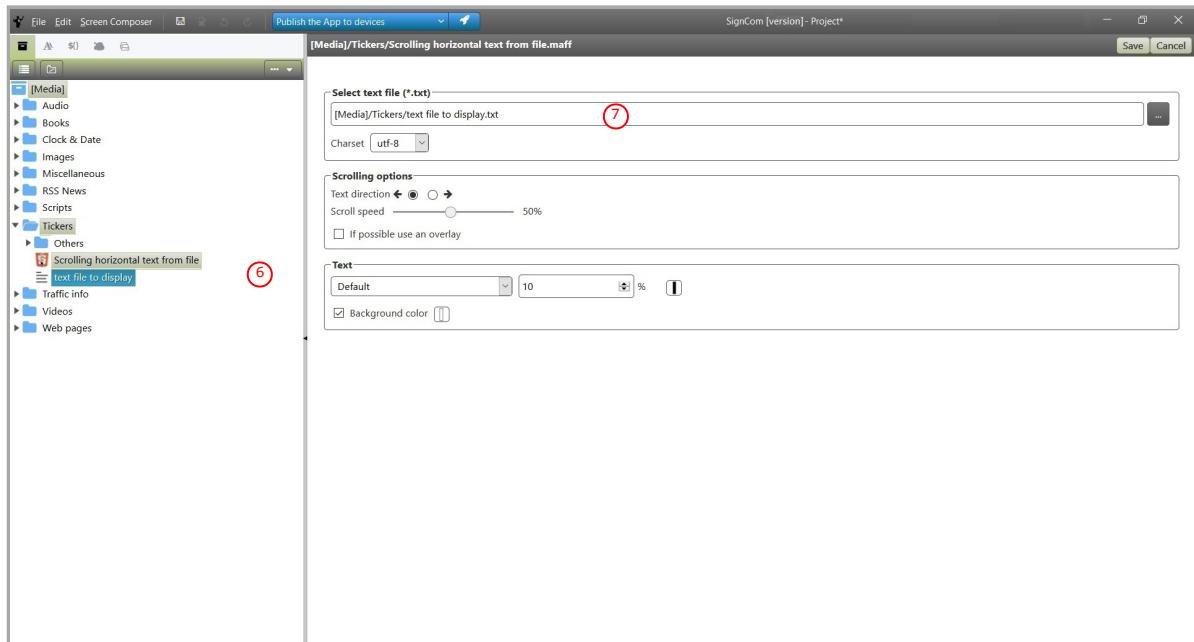


☞ The available category values and ratio values are depending on the installed content models.

Enter a label for the HTML widget then click on the **ok** button.

☞ Each content model has its own form and requires that some inputs must be filled properly. In the example, the *scrolling text from file* content model requires a .txt file as input. So create a .txt file having as content the text to display.

Drop a text file from the **Media** tab (6) to the **Select a file (*.txt)** (7) input.



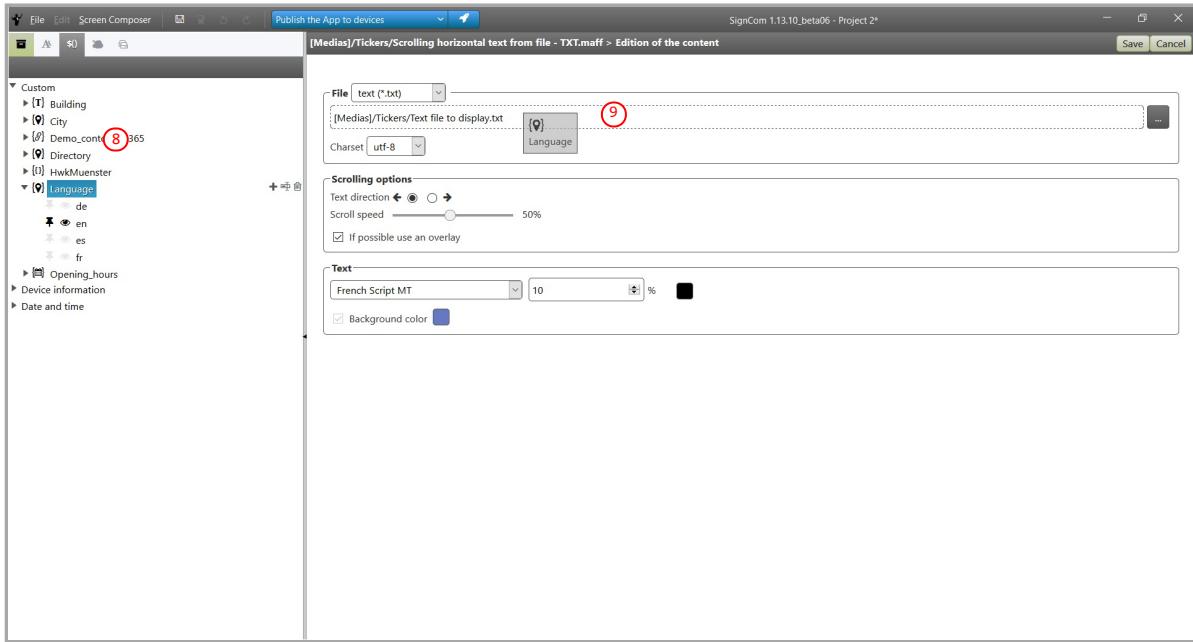
Click on the **Save** button to save your modifications.

- ☞ The **images** input of the recent content model form should support the drag'n drop of images coming from **Medias** tab.
- ☞ The **font** input of the recent content model form should support the drag'n drop of fonts coming from **Fon**ts tab.

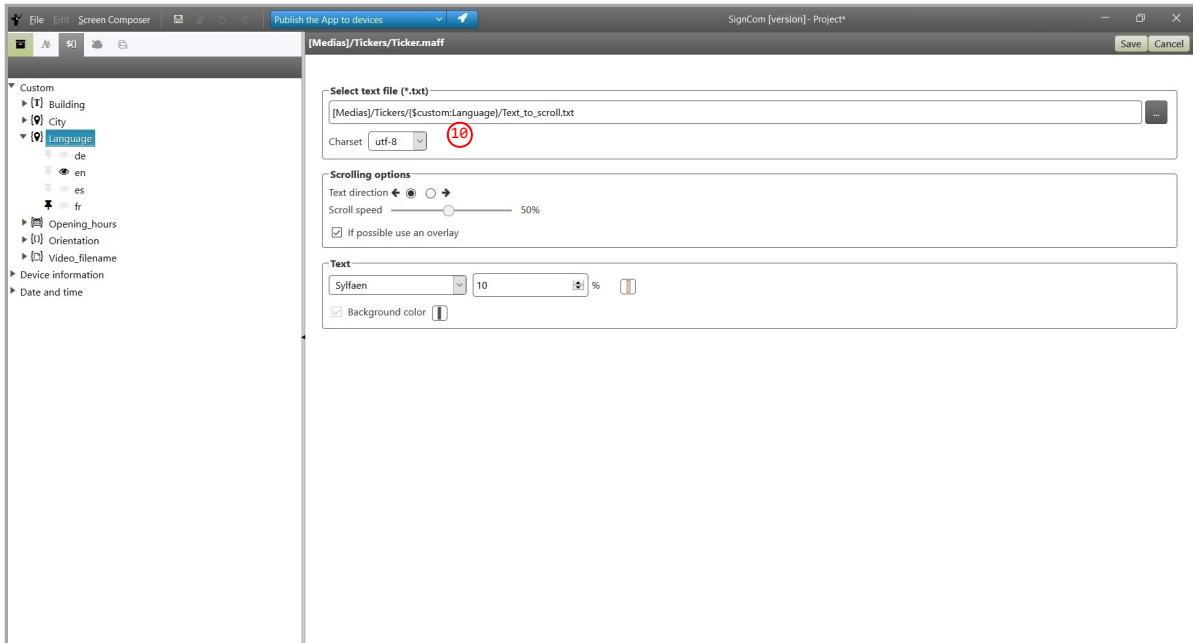
Variables

The URL input and the text input of the recent content model form should support drag'n drop of `Text` variables name and custom `localized` variable names coming from `Variables` tab.

Edit the content model form (*Ticker*), click on the `Variables` tab (8), and drop a custom `localized` variable name (e.g. *Language*) to the URL input (9) of the content model form.



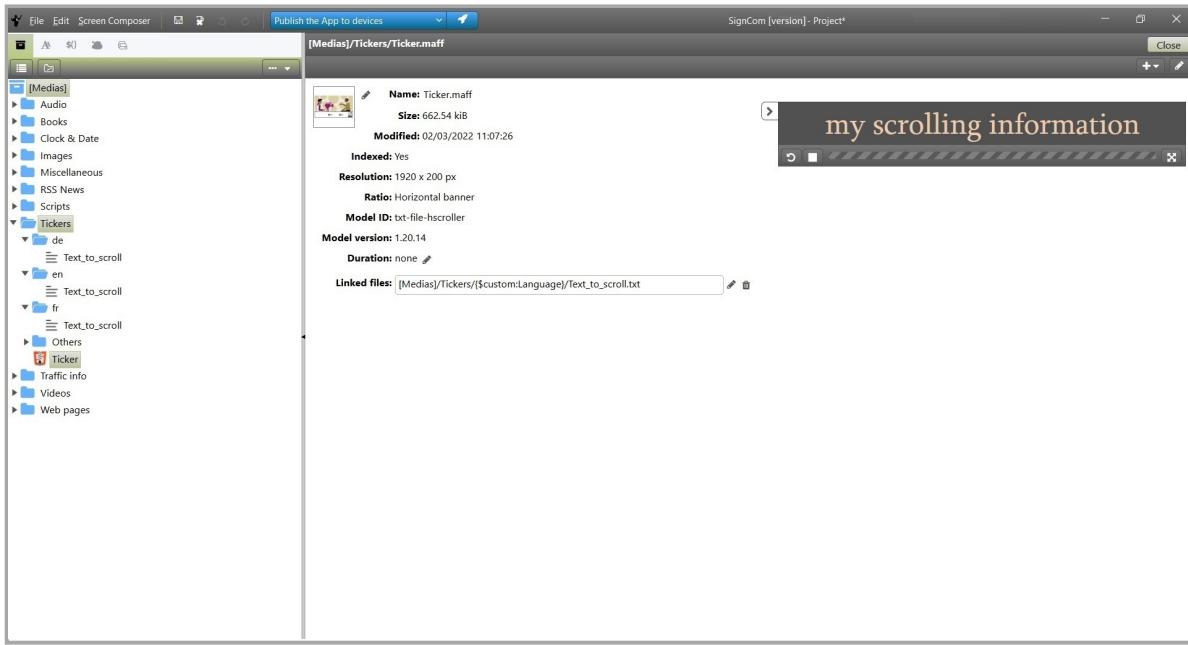
This piece of text with this syntax `${custom:<variable_name>}` (10) (e.g. `${custom:Language}`) appears in the URL input of the content model form after the drop.



Save the form. Arrange to have the suitable folders and files in your `Medias` tab:

- `/en/Text_to_scroll.txt`,
- `/fr/Text_to_scroll.txt`,
- `/de/Text_to_scroll.txt`.

Then launch the preview of the widget.

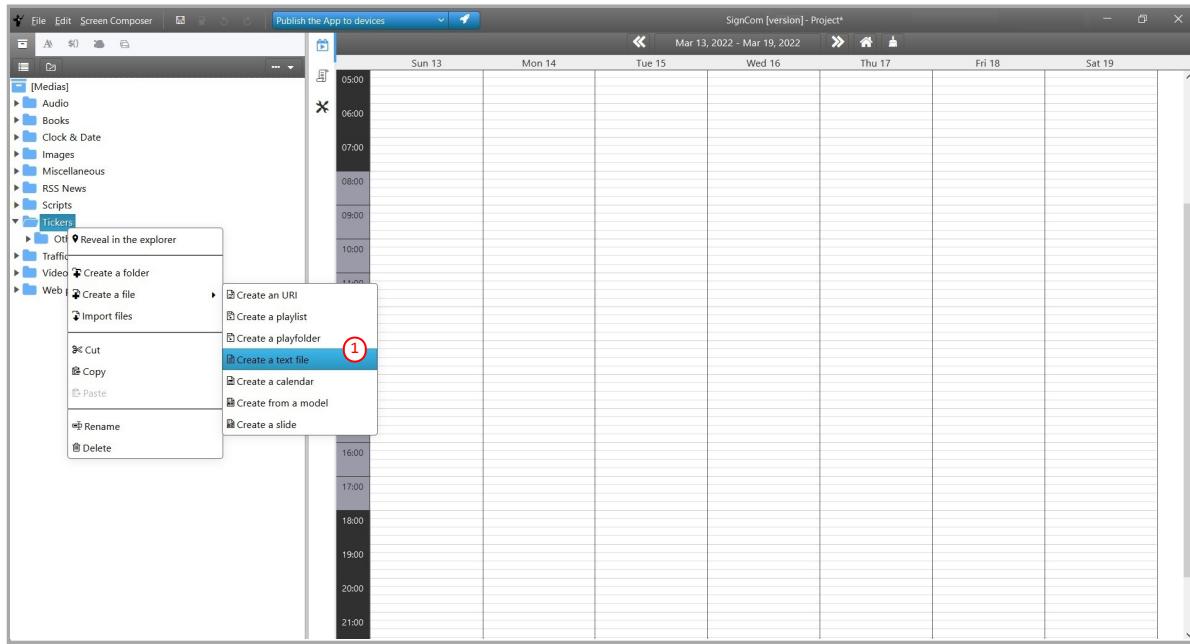


☞ The variable input of some specific content model form should support a variable picker to load `calendar` variable name or `json` variable name.

3.2.7 Medias > Text file

The text files are the files having these file extensions: `.css`, `.csv`, `.htm`, `.html`, `.ics`, `.js`, `.json`, `.md`, `.mdp`, `.sdp`, `.srt`, `.tsv`, `.txt`, `.vtt`, `.xml`.

To create a text file, select a folder in your library then in the contextual menu of the folder, select the `Create a file` item then click on the `Create a text file` **(1)** button.

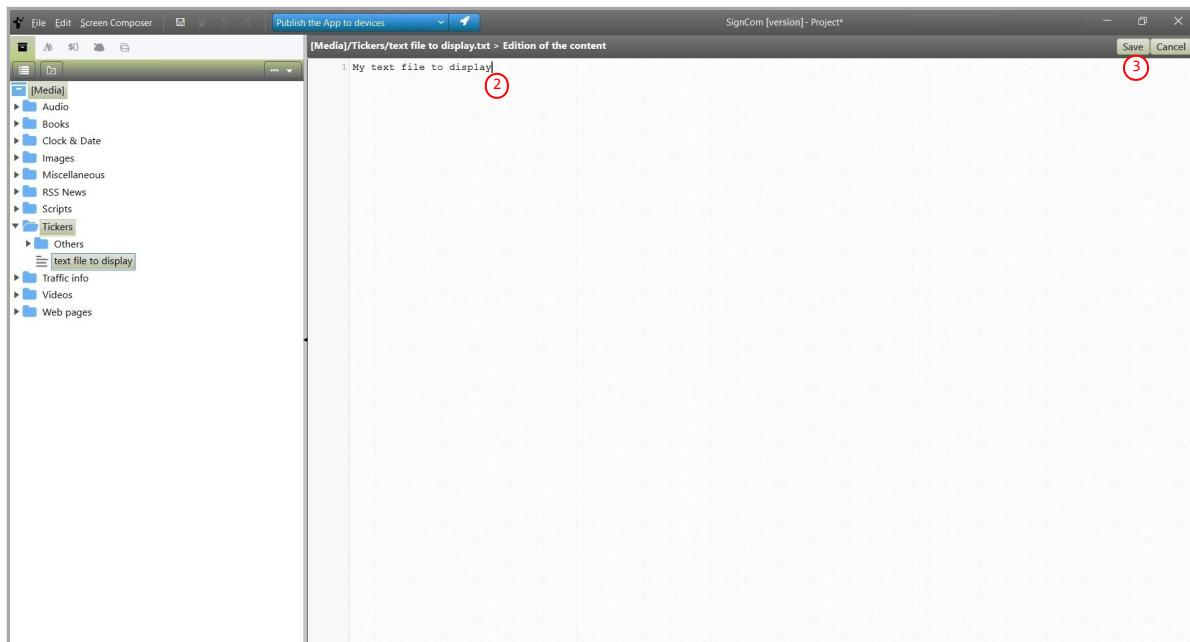


Select the appropriate extension according to your needs. Update the text file **(2)** content and click on the `Save` **(3)** button.

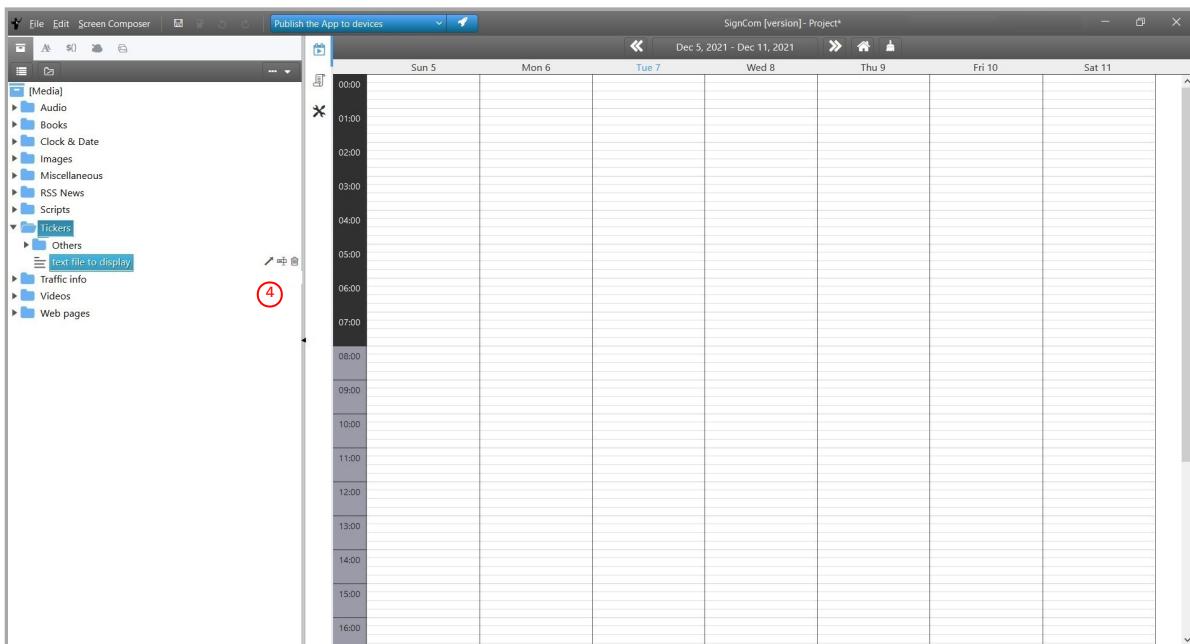
⚠ When importing text files such as `.js`, `.htm`, `.html`, make sure they are in UTF-8 format first.

☞ Depending on the chosen file extension, check whether some red timestamp is displayed in the file showing the probable syntax error.

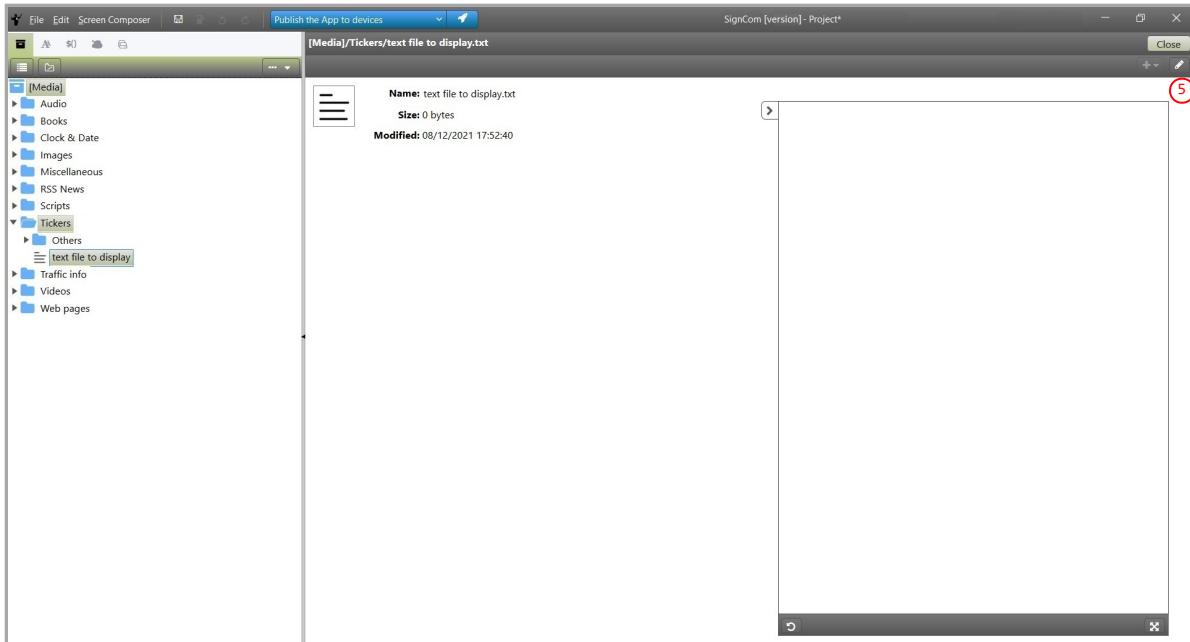
☞ The `.ics` files created with `Create a text file` cannot be recognized by the calendar editor. In this case, the `.ics` file is simply opened in raw mode. In case you want that your `.ics` file is recognized by the calendar editor, create it with the `Create a calendar` item. For further information, refer to the chapter § [Medias > Calendar](#)



To edit a text file, select the text file in the library and click on the Open  (4) button.



Then click on the Edit the content  (5) button.



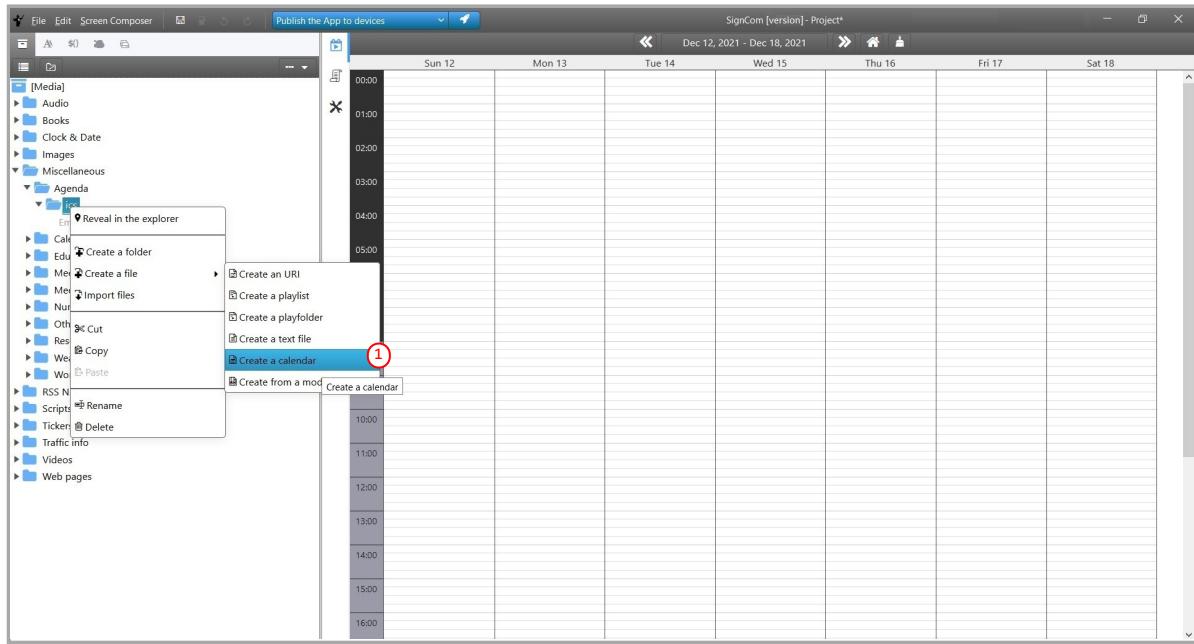
Modify the content according to your needs and click on the Save  (4) button.

If required, click on the close button to close the file and make appear back the SignCom App configuration form.

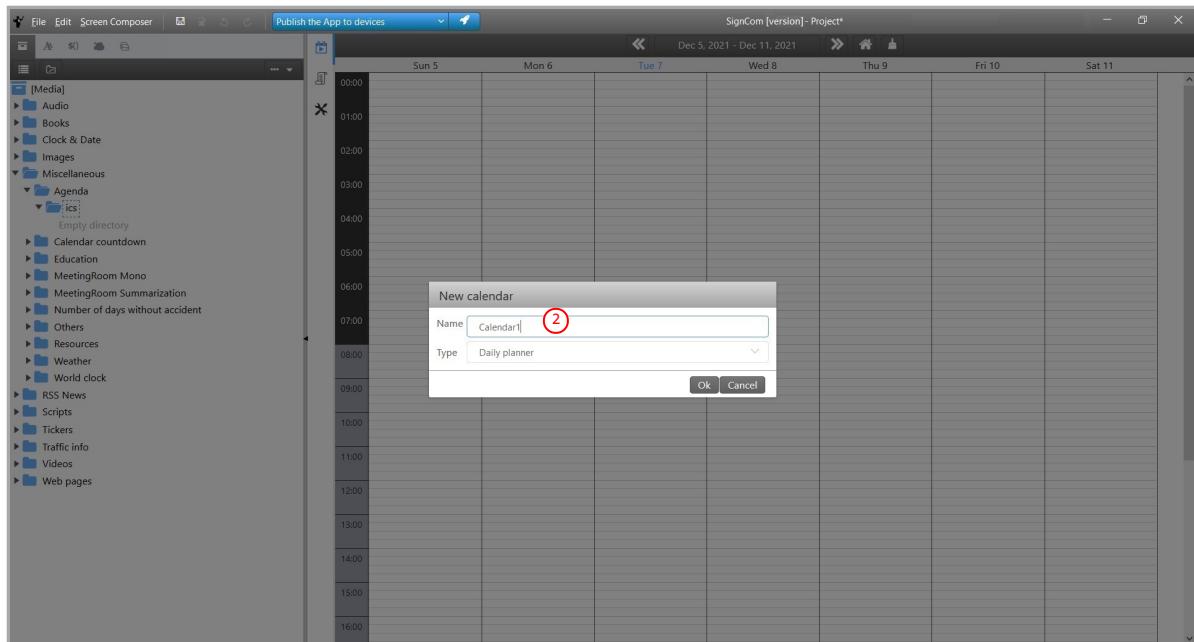
3.2.8 Medias > Calendar

A calendar is an `.ics` file. The build-it calendar editor supports a form that allows to simply modify and add new events.

Select a folder and right click on it to display its contextual menu. Select the Create a file item then click on the Create a calendar (1) item.



Enter a label for your calendar (e.g. `Calendar1` (2)) and click on the `OK` button.



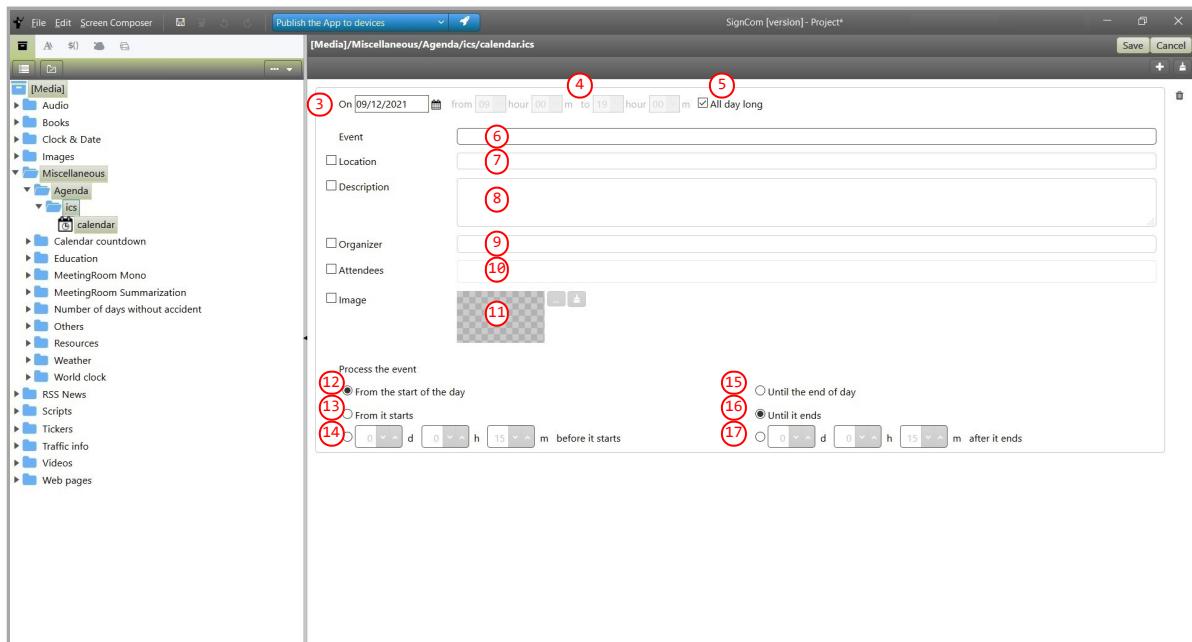
When being edited, the `calendar` is automatically recognized by the calendar editor which is able to list inside a form all the events contained in the `.ics` file from the older event at the top of the form to the newer event at the bottom of the form.

In the form, fill in the attributes of your event:

- **③** date,
- **④** start time/end time or **⑤** All day long ,
- **⑥** label for the Event ,
- **⑦** Location of the event,
- **⑧** Description of the event,
- **⑨** Organizer of the event,
- **⑩** Attendees having to attend the event,
- **⑪** Image **1**: allows to attach an image to the event.

1 Use the file picker to add an image. The drag'n drop of an image from the medias library to the image checkerboard **⑫** is supported.

2 The `Calendar` editor form can be only available when the `.ics` has been created with INNES SCreen Composer G4 with the Create a calendar button.

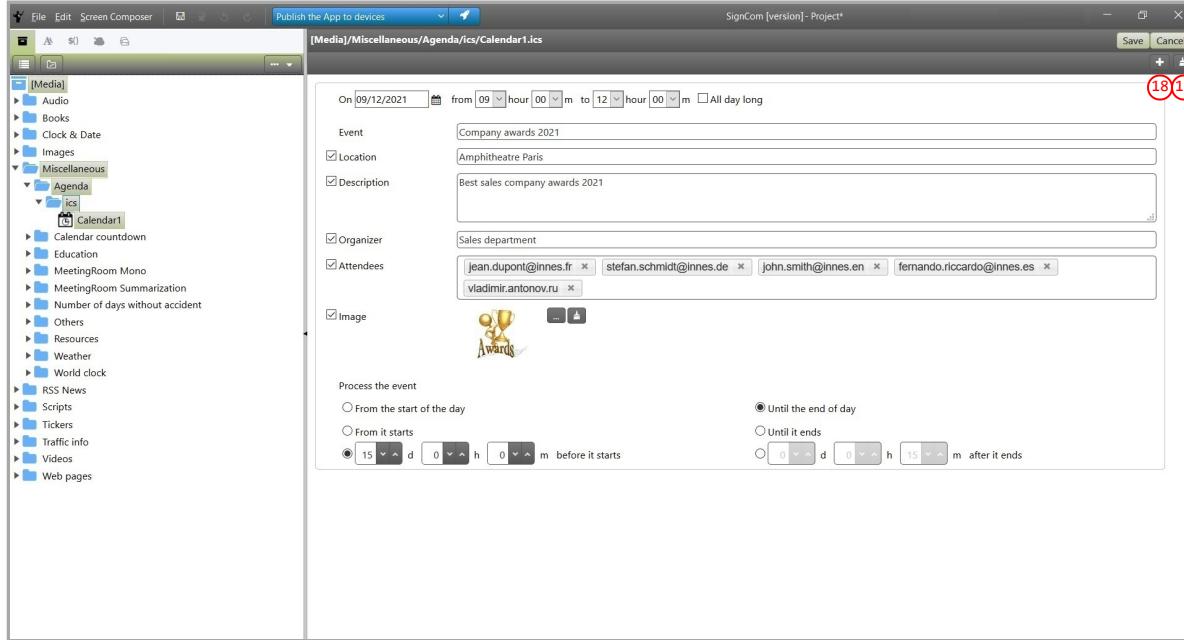


It is possible to set event alarms to be able to anticipate the display of a future event using the options in **Process** the event .

- **(12)** From the start of the day : the event can be displayed from 00:00:00, the current day of the event,
- **(13)** From it starts : the event can be displayed as soon as it has started,
- **(14)** <x> d <y> h <z> m before it starts : the event can be displayed until <x> d <y> h <z> m before it starts.

With the same option, it is possible to set alarm to make that the display of an event persists even when it has ended.

- **(15)** Until the end of the day : the event can be displayed until 23:59:59, the current day of the event,
- **(16)** Until it ends : the event can be displayed while it has not ended,
- **(17)** <x> d <y> h <z> m after it ends : the event can be displayed until <x> d <y> h <z> m after is has ended.

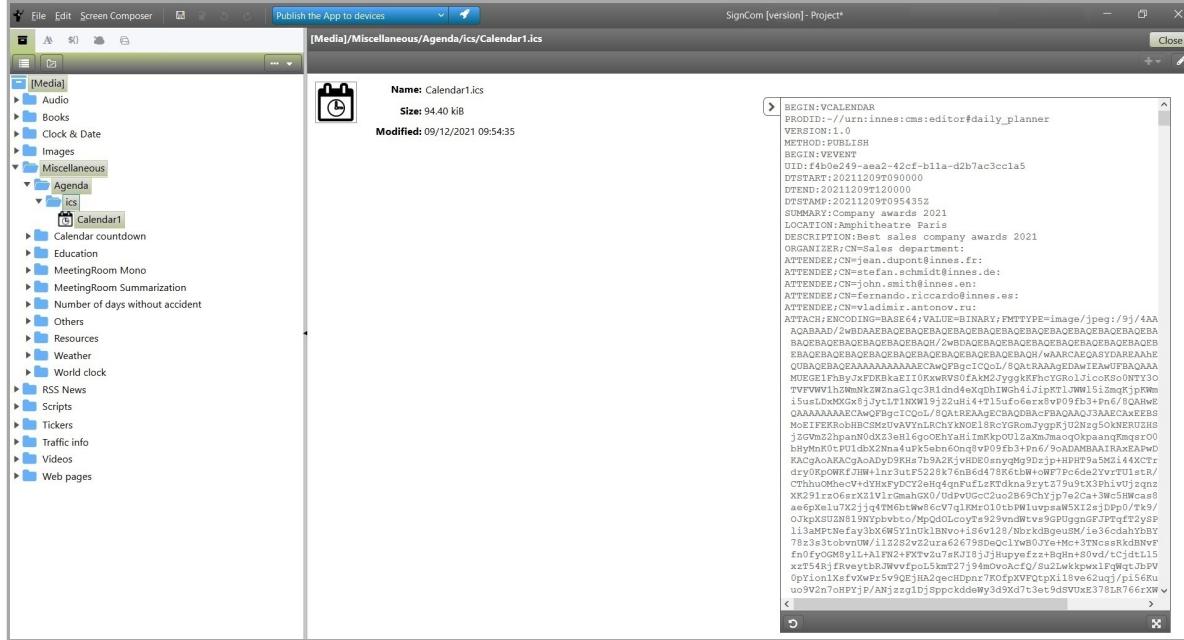


Add as many events as necessary with the button **Create an event** + **(18)**.

If required, click on the **Delete past events** **–** **(19)** button to delete all the past events from the **.ics** calendar file.

Click on the **Save** button.

Tip: You may have to scroll to the bottom watch all the event.

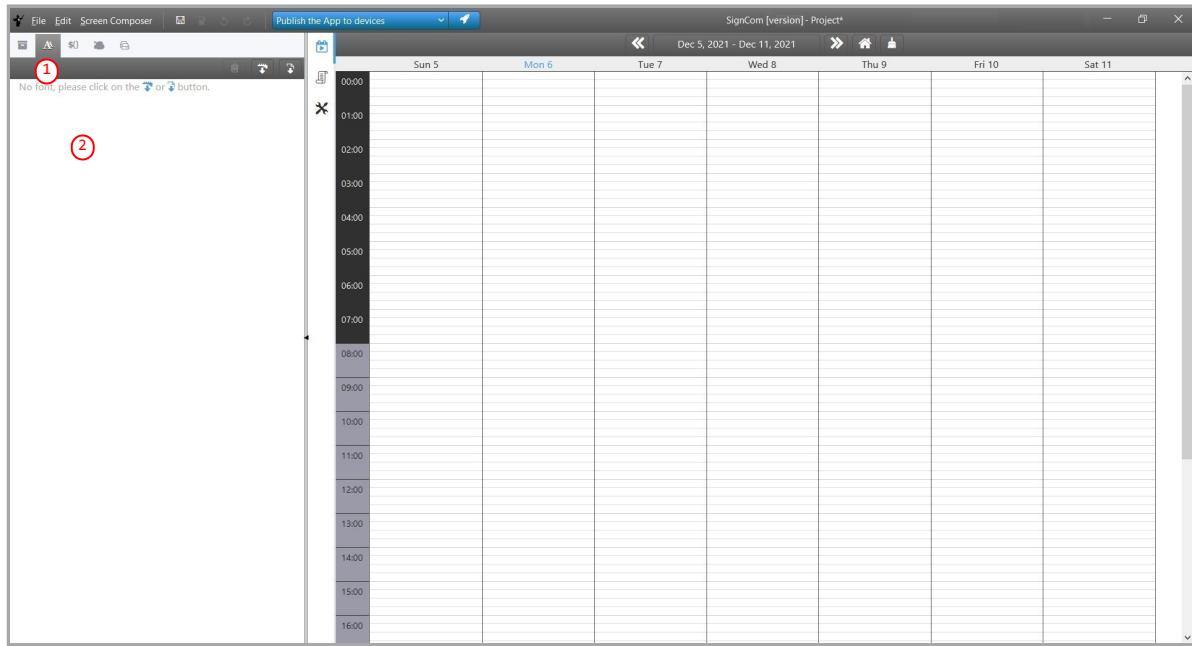


This calendar (.ics) can be directly loaded by dedicated HTML widget displaying events. For further information, refer to the chapter § [Medias > Create from a model](#).

3.3 Fonts

On the left side, click on the **Fonts A (1)** tab.

When no font face has been imported in your project, no font face is visible in the **Fonts (2)** tab.

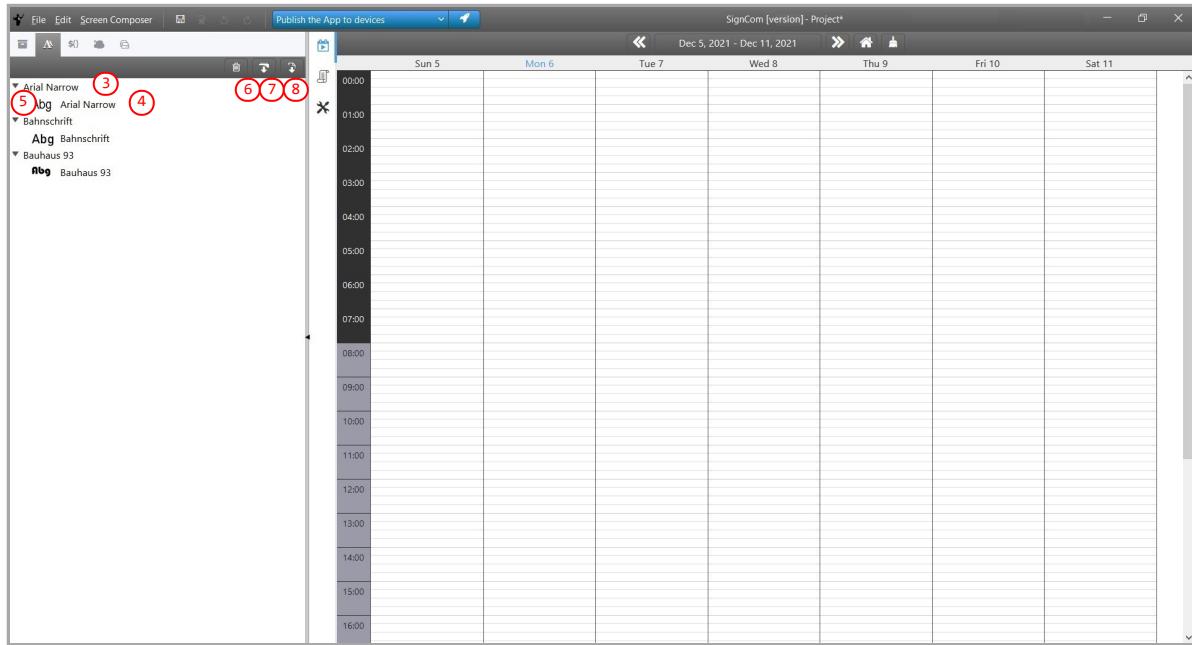


Click on the **Import fonts from the system** (7) button to import fonts from the MS-Windows system.

Click on the **Import fonts from a file** (8) button to import fonts from a MS-Explorer directory containing fonts files.

■ To be supported, fonts must be in .otf or .ttf format.

Once imported in your INNES Screen Composer G4 , the fonts names (3) are displayed with their font family (4). For each, The Abg text (5) is displayed as rendering example with the fonts style.



■ To remove several fonts at a time, select either several fonts names or several fonts families and click on the Remove (6) button.

⚠ To have a perfect document rendering, when installing a font, it is advised to install the maximum of declination of fonts belonging to the same font family.

3.4 Variables

The variables can be used in medias like `URIs` or `playfolders` to play, with a same Screen Composer project, a different content depending on:

- the variable values affected to the devices,
- the variable values intrinsic to the device,
- the system date of the device.

The variables type to use in these different cases are respectively:

- the `Custom` variables,
- the `Device information` variable,
- the `Date and time` variable.

The `Custom` variables name and the `Custom` variables values must be created then affected to the different device by the end user.

The `Device information` variable values must be created by the user in relation to the values existing in the fleet of devices.

The `Custom` variable values and the `Device information` variable values created by the user are linked to the repository in which they have been created. If a new project is created by using this same repository, these variable values previously created stay available.

3.4.1 Custom variable

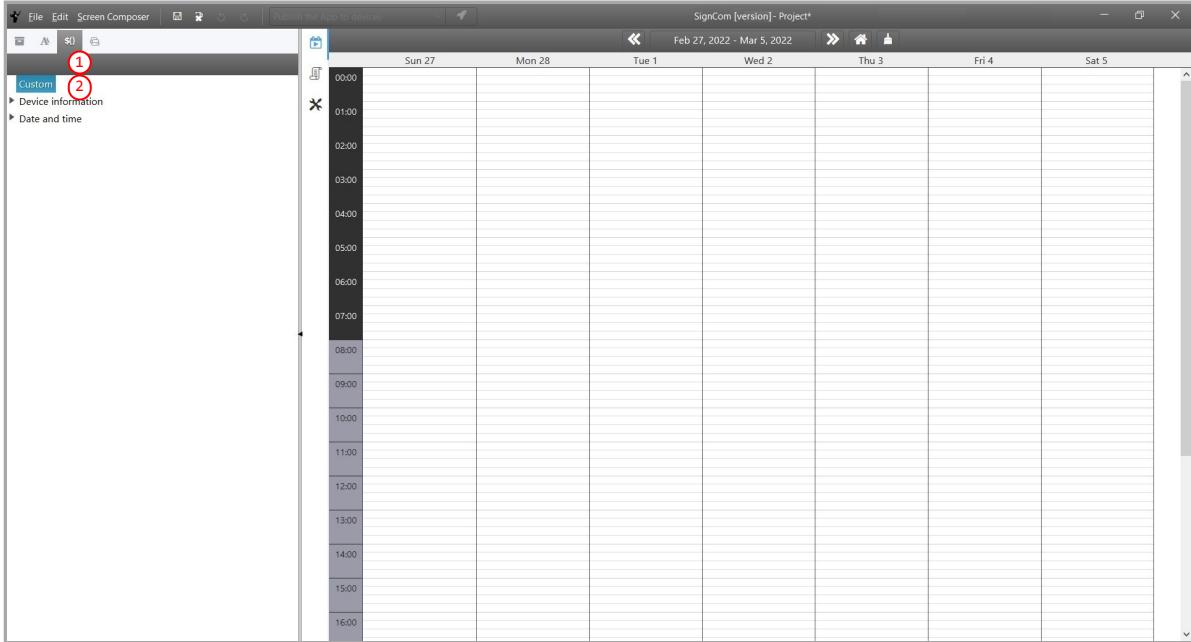
The `custom` variables consists in:

- a variable name,
- one or several values associated to the `custom` variable name.

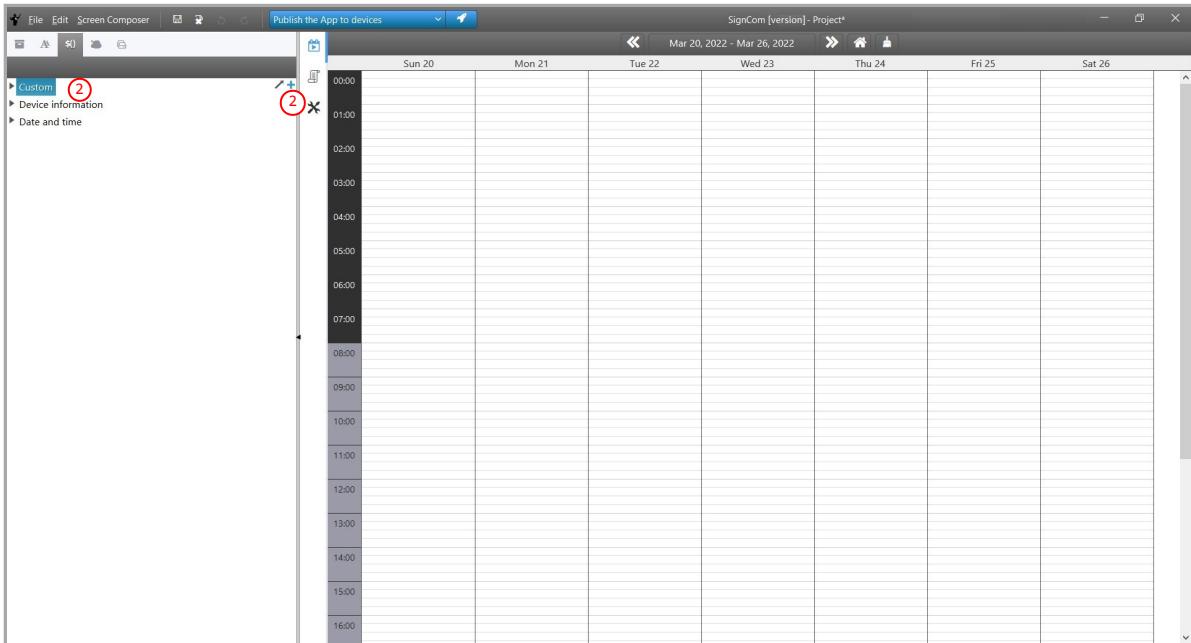
The supported characters for the `custom` variables names are: `_`, `0-9`, `a-z`.

When dropped in a URL, the `custom` variable is recognized with the syntax `${custom:<variable_name>}` .

To create new `custom` variable names, click on the `Variables`  **①** tab of the library then click on the `Custom`  **②** item.



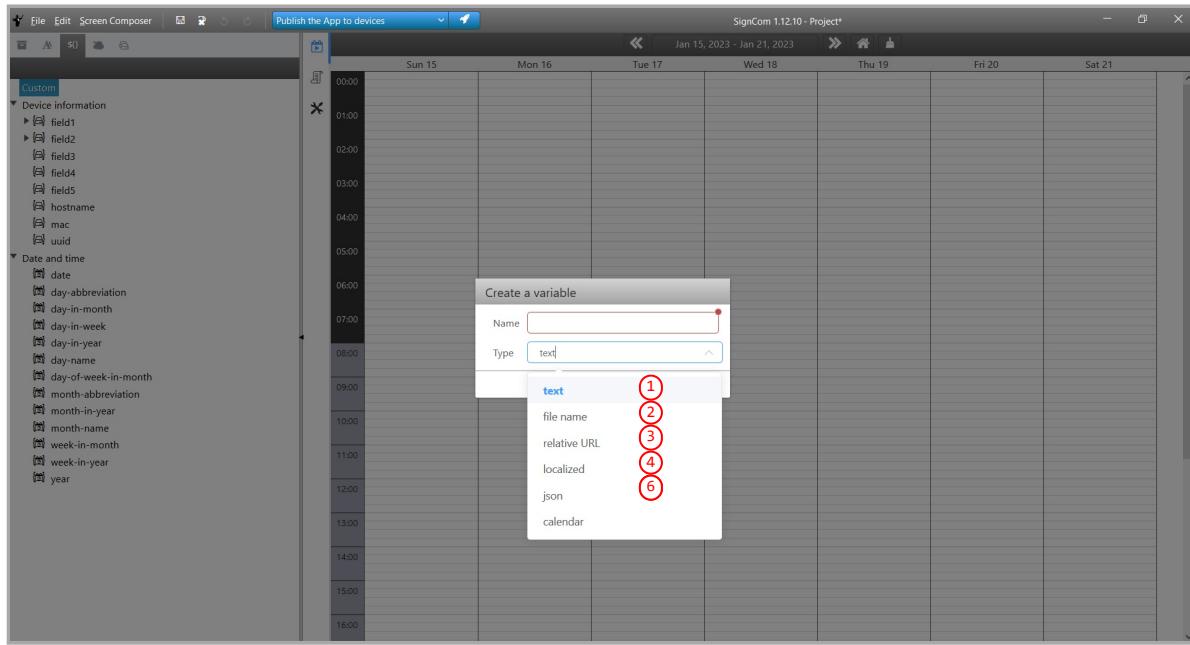
Then with your mouse, fly over the `Custom`  **②** item to make appear the  **③** button, then click on it.



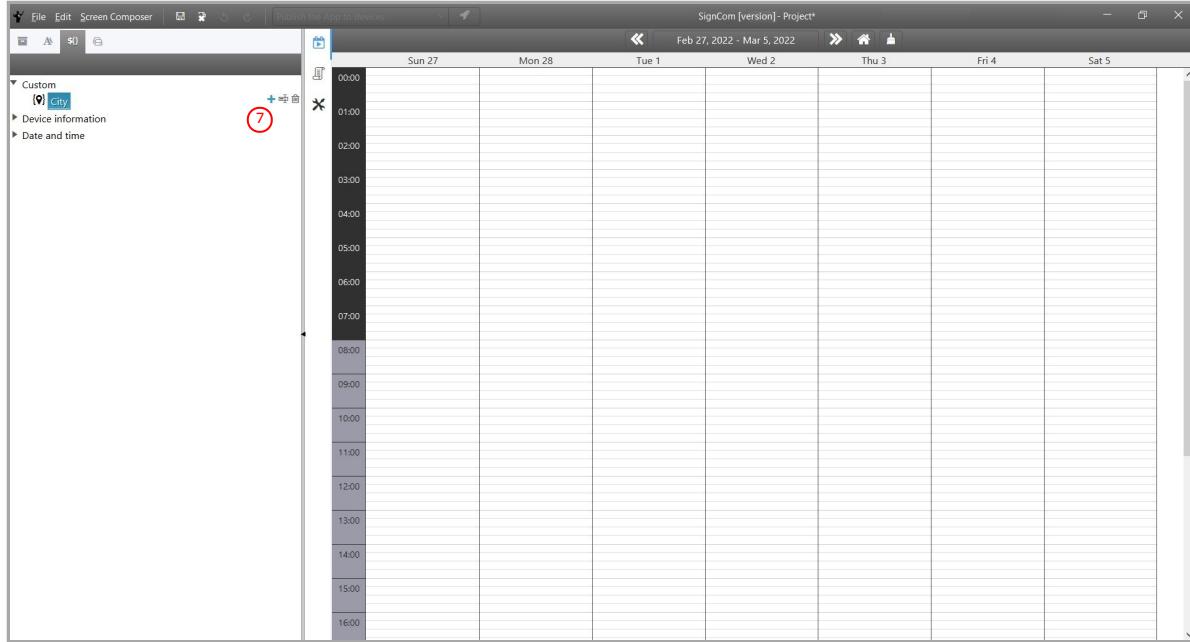
In relation with the variable you have to create, you need to choose among the different variables types:

- **①** text (*default selected type*),
- **②** filename ,
- **③** relative URL .
- **④** localized ,
- **⑤** json ,
- **⑥** calendar .

In the example, choose the `localized` **④** type. Then you need to enter a variable name (e.g. type `City`). Press on the OK button.



When the variable name is created (e.g. `City`), fly over it to make appear the **+ ⑦** blue button, then click on it to create new variable values.



Restart the operation as much as you have values to create.

It is the same procedure for all the custom variable types.

Custom variables of type custom

The value for `custom` variable of type `localized` (9) (1) has a characters string format.

This is an example of name and values for a `custom` variable of type `localized` (9).

name for a <code>custom</code> variable of type <code>localized</code> (9) (1)	values for a <code>custom</code> variable of type <code>localized</code>	values labels for a <code>custom</code> variable of type <code>localized</code>
City (2)	Paris (3) London (5) Berlin (7)	(4) (6) (8)
Lang	fr en de es	

The `custom` variables of type `localized` (9) can be used afterwards in the URL of `URI` medias, in the `playfolder` medias or in the URL or some inputs of the content model forms:

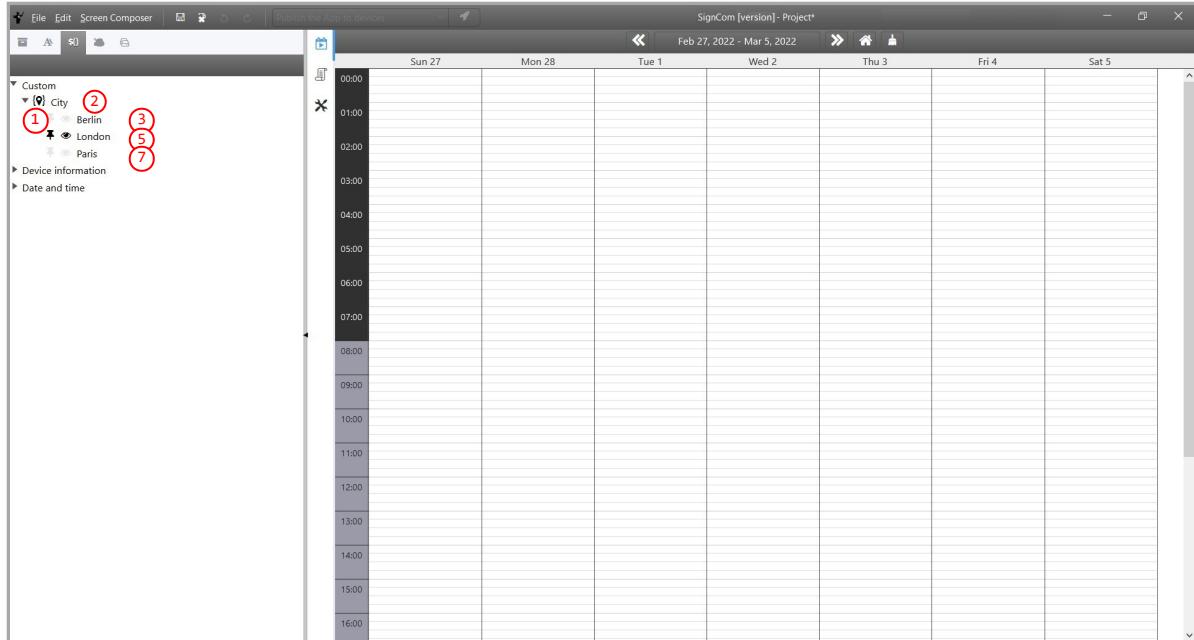
- example of a playfolder URL input containing the `City` (2) `custom` variable of type `localized` variable name:
 - [Medias]/presentation_{\$custom:City}.ptx
- example of URL input containing the `Lang` `custom` variable of type `localized` :
 - [https://www.innes.pro/{\\$custom:Lang}/](https://www.innes.pro/{$custom:Lang}/)

Tip The values for the `custom` variable of type `Localized` (9) can contain character coming from different alphabets.

When creating a new `custom` variable of type `localized` variable, select the `localized` type. Enter a name for your `localized` (e.g. `City` (2)).

Add the required values for the `custom` variable of type `localized` (9) (e.g. `Berlin` (3), `London` (5) then `Paris` (7)).

Tip The values labels (4), (6), (8) for `custom` variable of type `localized` are optional.



Tip The characters /, \, :, *, ?, ", <, >, | and . are not supported in values of the personalized variables of type `localized`.

Custom variables of type relative URL

The value for `custom` variable of type `relative URL` `{}` ① has a URL string format corresponding to a shared directory, without the scheme `https://` or `\smb://`, supporting in particular the following characters: `/`, `\`, `:`, `*`, `?`, `"`, `<`, `>`, `|` and `..`.

This is an example of name and values for a `custom` variable of type `relative URL` `{}`.

name for custom variable of type relative URL {} ②	values for custom variable of type relative URL	values labels for custom variable of type relative URL
<code>Sharepoint_url</code> ②	<code>contoso446-my.sharepoint.com/:f/r/personal/user1_contoso_com/</code> <code>..../Documents1?csf=1&web=1&e=Q7Mk9y</code> ③	<code>Sharepoint_url1</code> ④
	<code>contoso446-my.sharepoint.com/:f/r/personal/user1_contoso_com/</code> <code>..../Documents2?csf=1&web=1&e</code> ⑤	<code>Sharepoint_url2</code> ⑥
<code>SmbSharedDirectory_url</code>	<code>192.168.0.11/shared_directory/Documents1</code> <code>192.168.0.11/shared_directory/Documents2</code>	<code>SmbSharedDirectory_url1</code> <code>SmbSharedDirectory_url2</code>

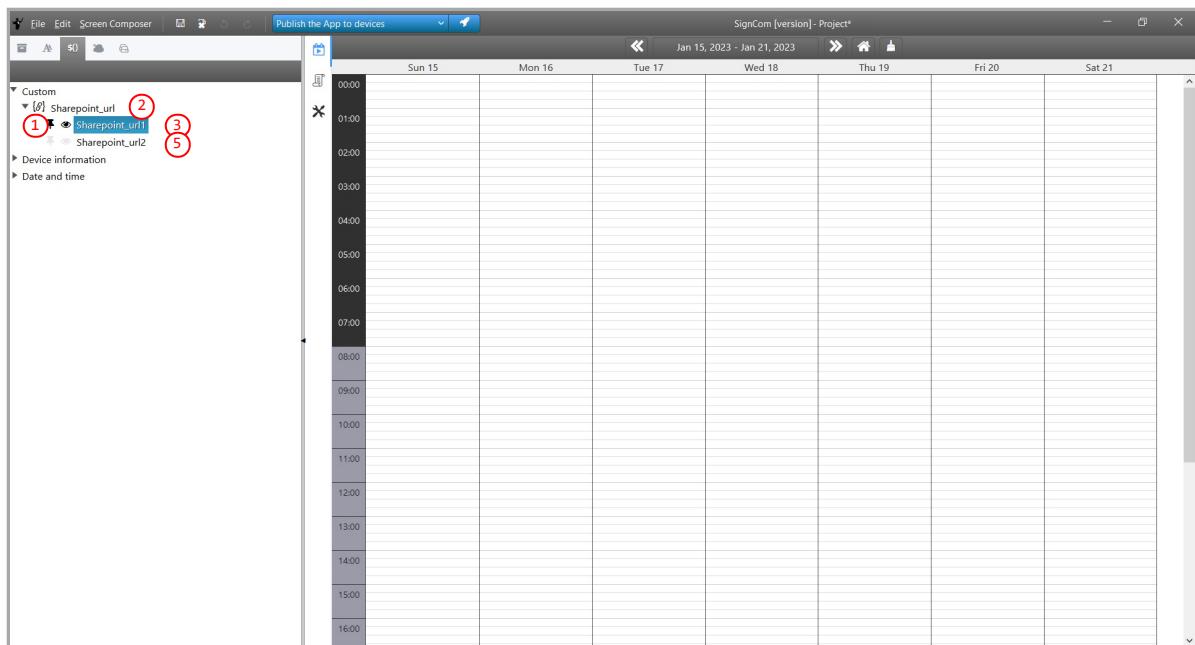
The `custom` variables of type `relative URL` `{}` can be used afterwards in the playfolder with *Microsoft 365* service accounts :

- example of a playfolder URL input containing the `Sharepoint_url` ② `custom` variable of type `relative`:
 - `https://{$custom:Sharepoint_url}`

When creating a new `custom` variable of type `relative URL`, select the `relative URL` type. Enter a name for your `relative URL` variable (e.g. `Sharepoint_url` ②).

Add the required values for the `custom` variable of type `relative URL` `{}` corresponding to your Sharepoint URL (e.g. `contoso446-my.sharepoint.com/:f/r/personal/user1_contoso_com/Documents1?csf=1&web=1&e=Q7Mk9y` ③ with the label `Sharepoint_url1`, `contoso446-my.sharepoint.com/:f/r/personal/user1_contoso_com/Documents2?csf=1&web=1&e` ⑤ with the label `Sharepoint_url2`).

☞ The values's labels ④, ⑥ for `custom` variable of type `relative URL` are optional.



Custom variables of type text

A value for a `custom` variable of type `text` `{T}` can contain a string with special characters.

This is an example of name and values for a `custom` variable of type `text` `{T}` (1).

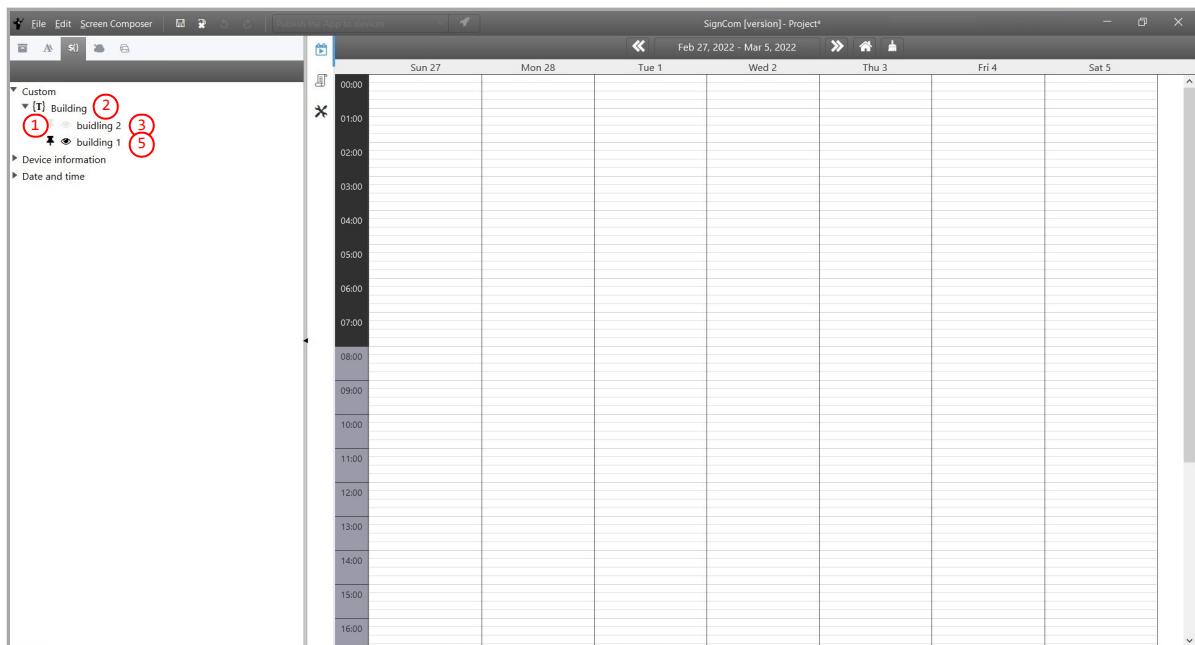
name for a <code>custom</code> variable of type <code>text</code> <code>{T}</code>	values for a <code>custom</code> variable of type <code>text</code>	values labels for a <code>custom</code> variable of type <code>text</code> .
Building (2)	Building 1 (3) Building 2 (5)	(4) (6)

The `custom` variables of type `text` `{T}` can be used afterwards in the text inputs of the content model forms:

- example of text input in a content model form containing the `Building` `custom` variable of type `text` `{T}`: `The today meetings for the ${custom:Buidling} building.`
- The `custom` variables of type `text` `{T}` cannot be used afterwards in the URL input of `URI` medias or in the URL of `playfolder` medias.*
- The values for `custom` variables of type `text` `{T}` can contain character coming from different alphabets.*
- The values label for `custom` variables of type `text` `{T}` are optional.*

When creating a new `custom` variable of type `text` `{T}`, select the `text` type. Enter a name for your `custom` variable of type `text` `{T}` (e.g. `Building` (2)).

Add the required values (e.g. `building 1` (3) and `building 2` (5)).



Custom variables of type filename

The values for `custom variables of type file name` (1) are composed of a string as filename followed by a file extension. The file extensions supported by the values of the `custom variables of type file name` are:

- the media extensions: .apng, .ASF, .avi, .gif, .htm, .html, .ismc, .jpg, .jpeg, .m2v, .m4a, .m4v, .maf, .maff, .mjpg, .mka, .mks, .mkv, .mov, .mp3, .mp4, .mpd, .mpg, .mpeg, .mxaf, .odt, .odp, .oga, .ogg, .ogv, .pdf, .png, .pps, .ppsx, .ppt, .pptx, .ps, .sdp, .svg, .ts, .vob, .webm, .webp, .wgt, .wma, .wmv,
- the resources extensions: .css, .csv, .htm, .html, .ics, .js, .json, .md, .mdp, .sdp, .srt, .tsv, .txt, .vtt, .xml.

This is an example of name and values for a `custom variable of type file name`.

name for a <code>custom variable of type file name</code> (2)	values for a <code>custom variable of type file name</code>	values labels for <code>custom variable of type file name</code>
Video_filename (2)	production1.mp4 (3) production2.mp4 (3) production3.m2v (7)	(4) (6) (8)
Calendar_resource	calendar1.ics calendar2.ics	

The `custom variable of type file name` (1) can be used afterwards in the URL of `URI` medias or in the URL of the content model forms:

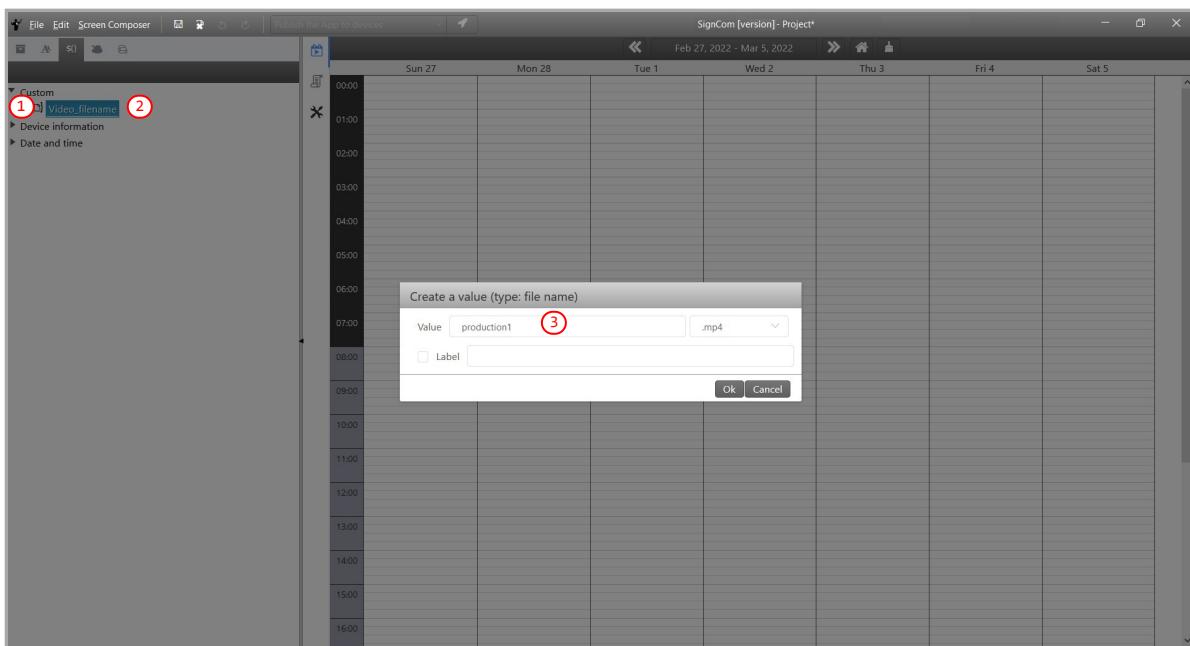
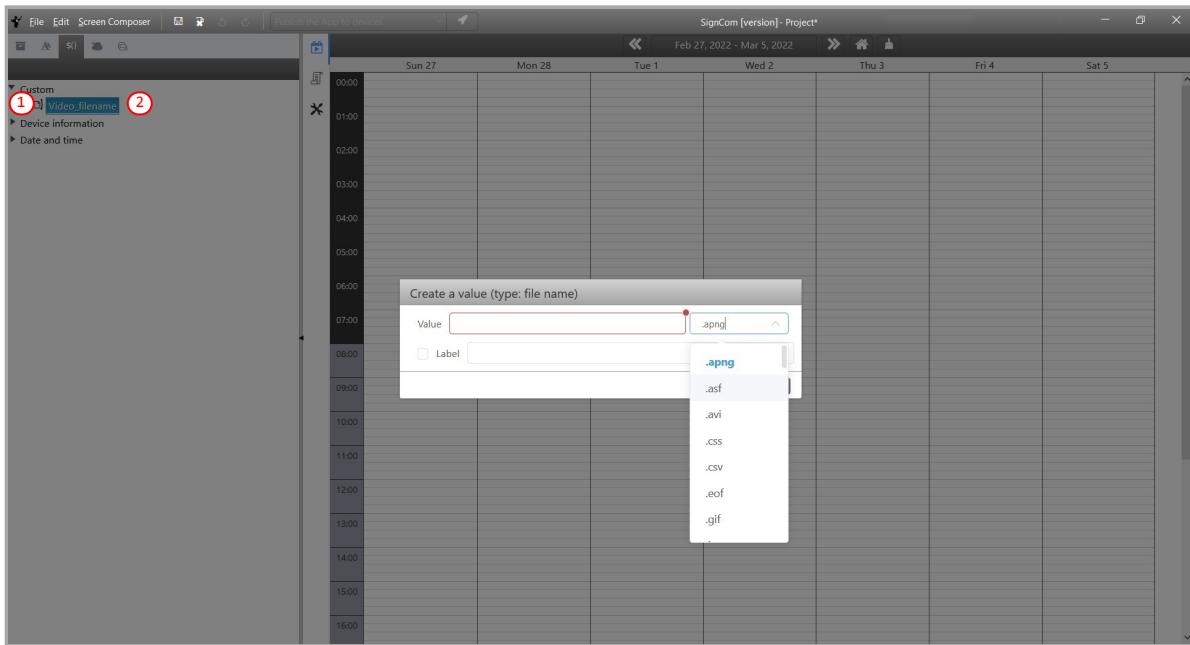
- example of an URL input containing the `Video_filename` `custom variable of type file name` (1): - [Medias]/video/{\$custom:Video_filename}
 - example of URL input containing the `Calendar_resource` `custom variable of type file name` (1):
 - [https://domain1/dir/{\\$custom:Calendar_resource}](https://domain1/dir/{$custom:Calendar_resource}).
- ☞ The values for `custom variable of type file name` can contain characters coming from different alphabets.
- ☞ The labels (4), (6), (8) for `custom file name` variable values are optional.

When creating a new `custom variable of type file name` (1), select the `file name` type. Enter a name for your `file name` variable (e.g. `Video_filename`).

Add the required values for the `custom variable of type file name` (e.g. `production1.mp4`) by filling properly the `Value` field then selecting the file extension among those suggested.

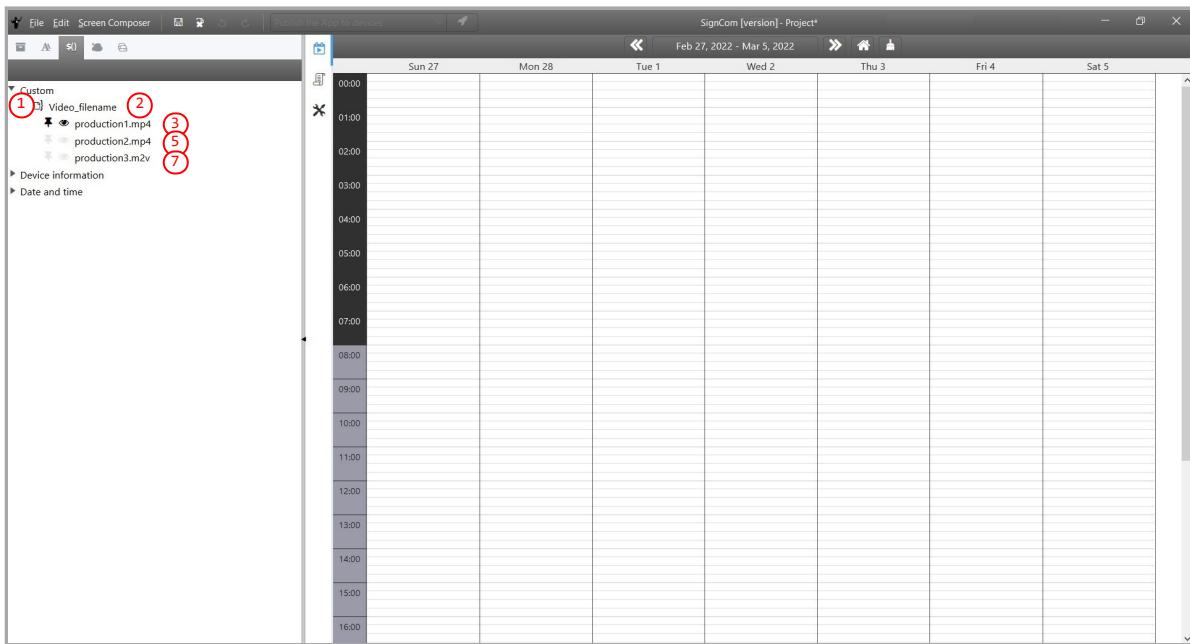
☞ Scroll to the bottom to view all the other available extensions.

This is an example of values for a `custom` variable of type `file name` .



Restart the operation to add other values for the `custom` variable of type `file name`.

This is a summary of the name and values for a `custom` variable of type `file name` .



Custom variables of type calendar

A `custom` variable of type `calendar` (1) is composed of:

- one or several day periods,
- with or without a day recurrence and,
- with one of several hours slots.

This is an example of name and values for a `custom` variable of type `calendar` (2).

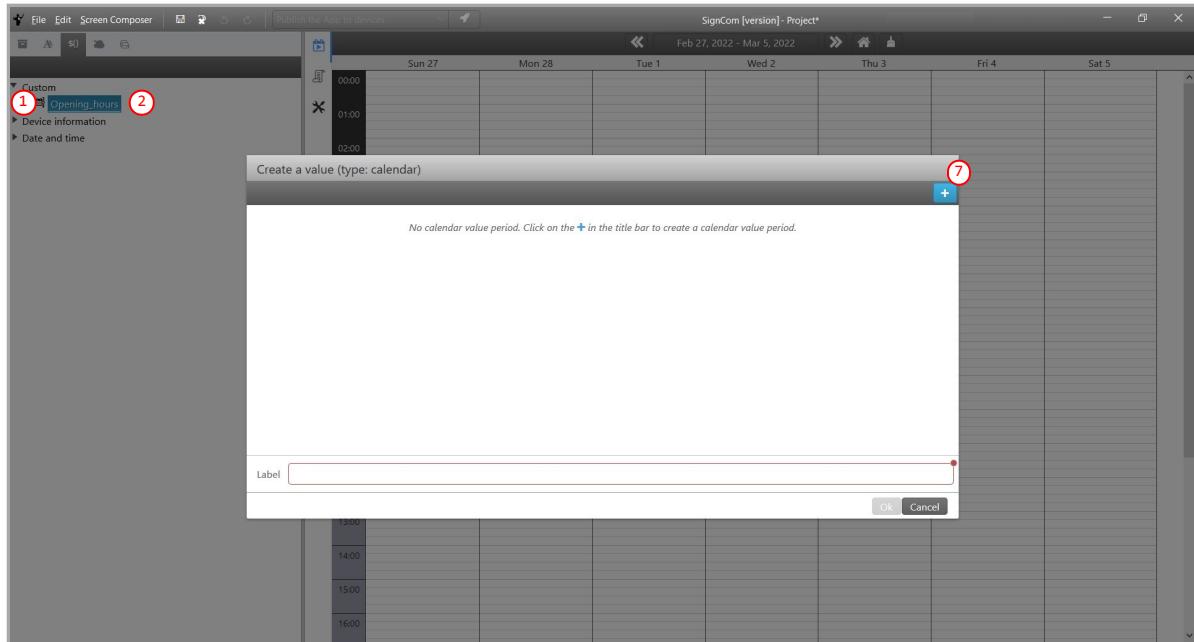
<code>name for a custom variable of type calendar</code> (3)	<code>values for a custom variable of type calendar</code> (4)	<code>values labels for custom variable of type calendar</code> (5)
(2) <code>Opening_hours</code>	<p>(3) from 1/07/2022 to 31/07/2022, every monday, tuesday, thursday, friday, from 10:00 to 12:00</p> <p>(5) from 1/01/2022 to 30/06/2022, every monday, tuesday, wednesday, thursday, friday, from 9:00 to 12:00 and from 14:00 to 19:00</p>	<p>(4) <code>opening_hours_july</code></p> <p>(6) <code>opening_hours_standard</code></p>

The `custom` variables of type `calendar` (2) can be used afterwards in some specific input of some specific content model form able to support such variable (e.g.: Calendar countdown content model).

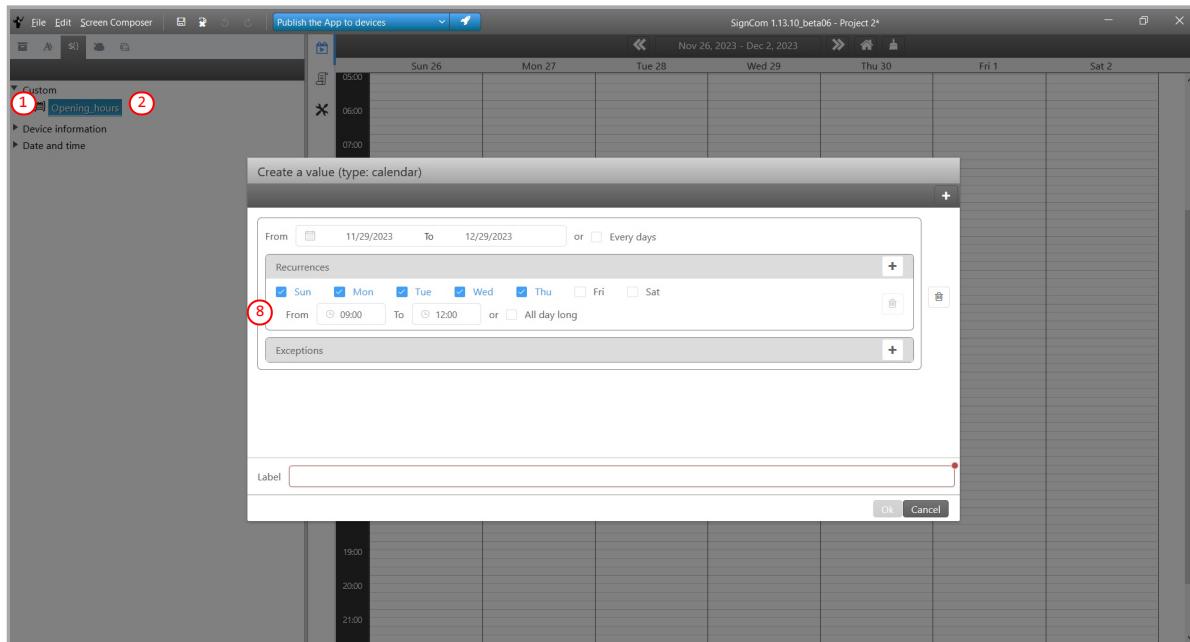
Note: The values labels for `custom` variable of type `calendar` (4) are mandatory.

When creating a new `custom` variable of type `calendar`, select the `calendar` type. Enter a name for your `custom` variable of type `calendar` (2) (e.g. `Opening_hours`). Then start to add values.

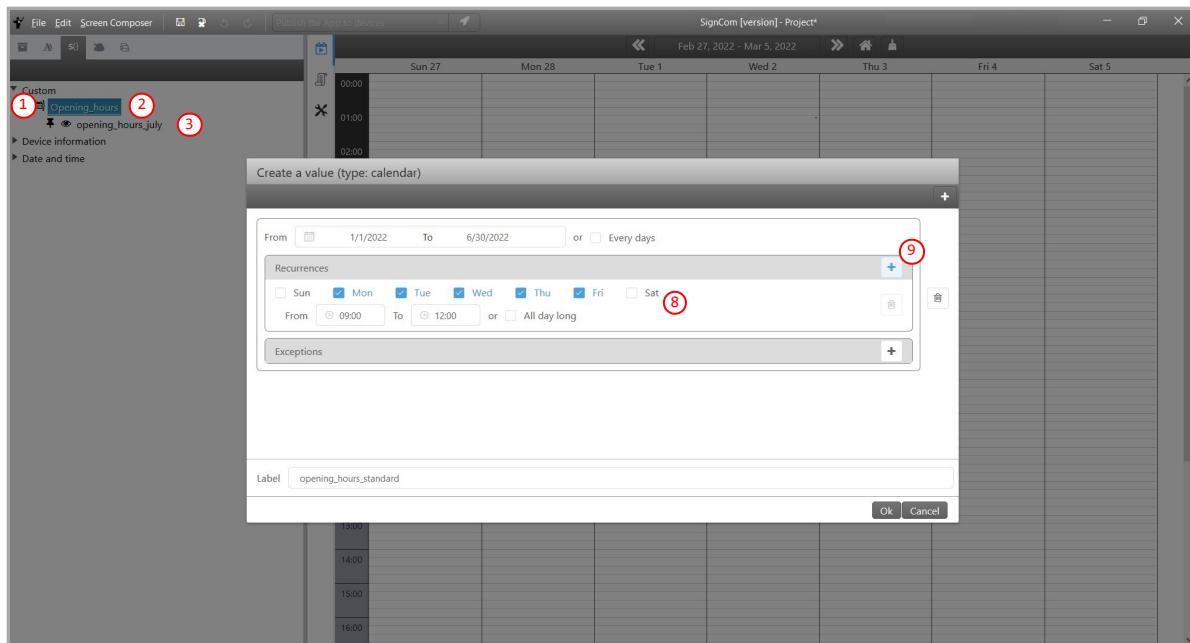
Press on the Add a calendar period (7) + button to add a calendar period .



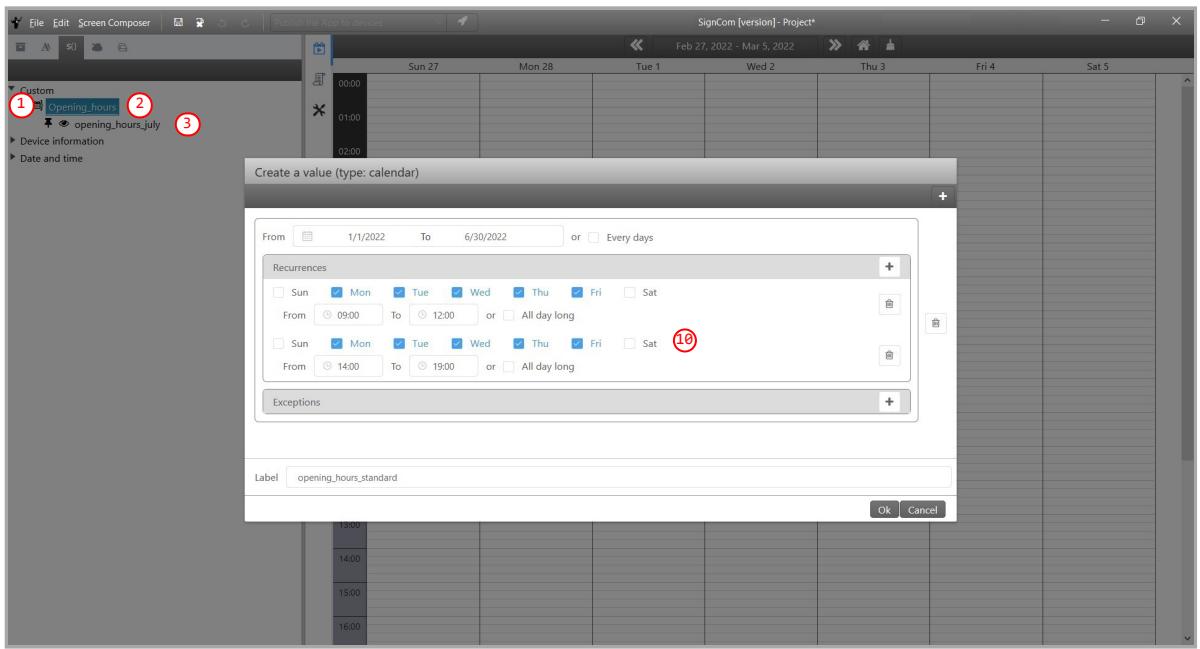
Fill the days recurrence and the timeslot (e.g. 9:00 - 12:00) ⑧.



Then press on the Add a recurrence ⑨ + to split the day in two timeslots.



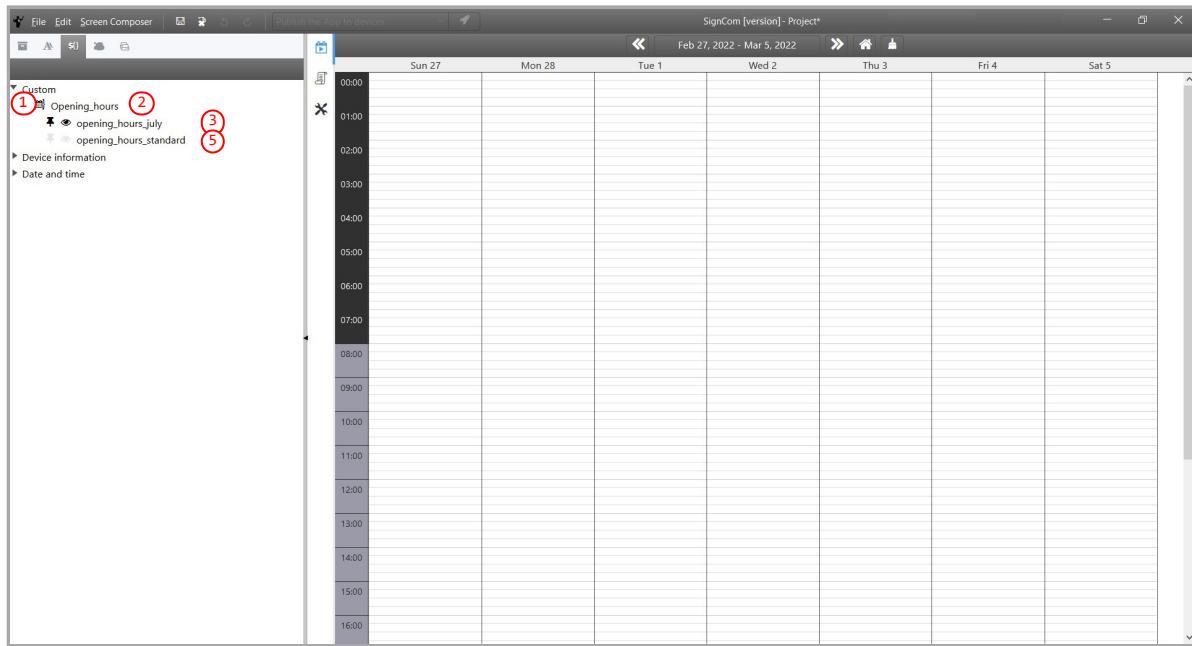
Fill the recurrences for the other timeslot (e.g. 14:00 - 19:00) ⑩.



Once done, enter a label (e.g. `opening_hours_standard`) then press on the OK button.

Restart the operation to add other values for the `custom` variable of type `calendar`.

This is a summary when all the required values for the `custom` variable of type `calendar` are created.



Custom variables of type json

A value for a `custom` variable of type `{0} json` has a content that could be found as well in a `.json` file.

*Consequently, the content of the values for the `custom` variable of type `{0} json` **(1)** must match the syntax of `.json` files.

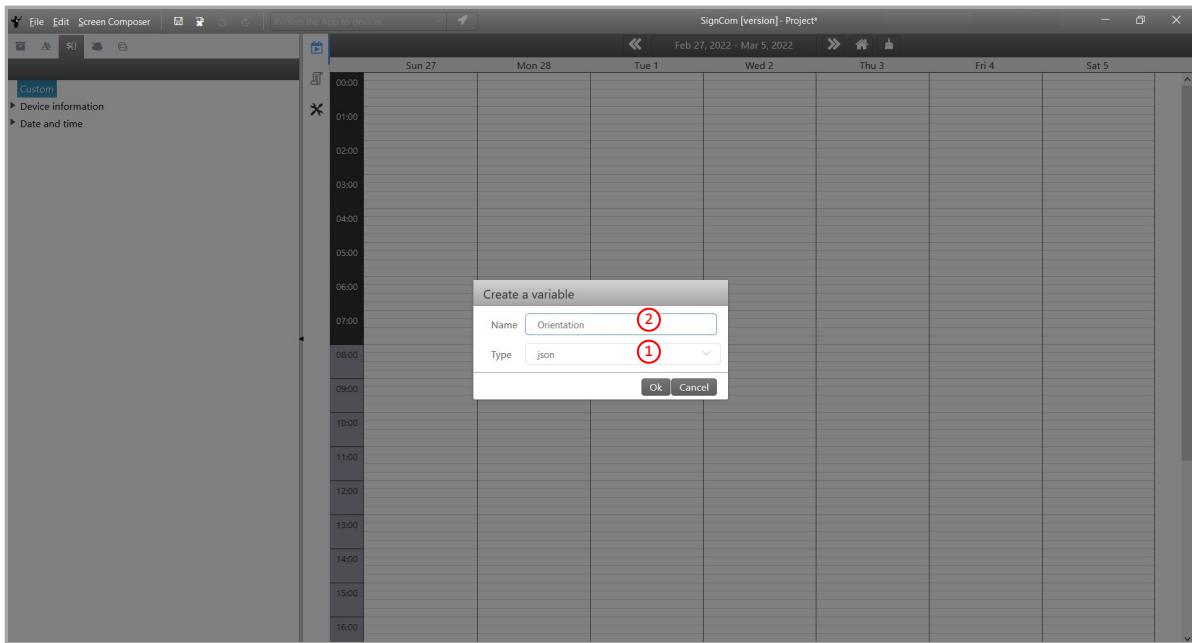
The `custom` variables of type `json` **{0}** can be used afterwards in some specific input of some specific content model forms able to support such variable type (e.g.: MeetingRoom - Summarization content model).

name for custom variable of type json {0} (1)	values for custom variable of type json	values labels for custom variable of type json
(2) Orientation	<pre>(3) [{"locations": [{"resource": "Room A-10", "sign1": "picto/escalatorDown.png"}]} (5) [{"locations": [{"resource": "Room B-20", "sign1": "picto/arrowRight.png", "sign2": "picto/escalatorUp.png"}]}]</pre>	(4) Orientation Room A-10 (6) Orientation Room B-20

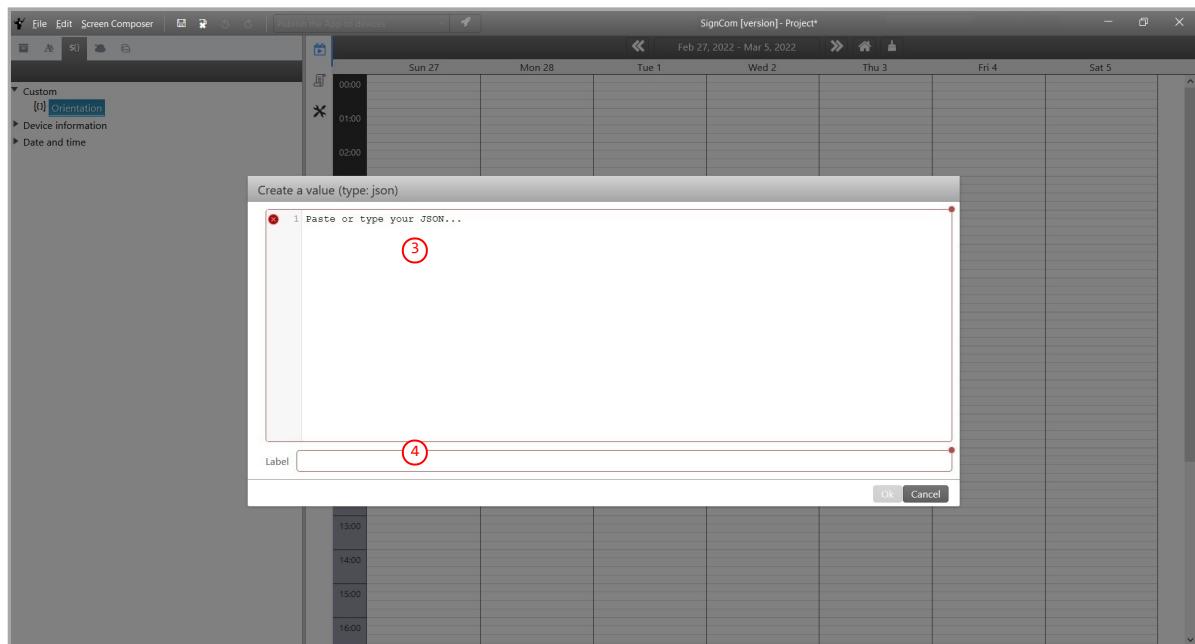
The `custom` variables of type `json` **{0}** can be used afterwards in some specific input of some specific content model form able to support such variable (e.g.: MeetingRoom - Summarization content model).

The values label for `custom` variables of type `json` **{0}** are mandatory.

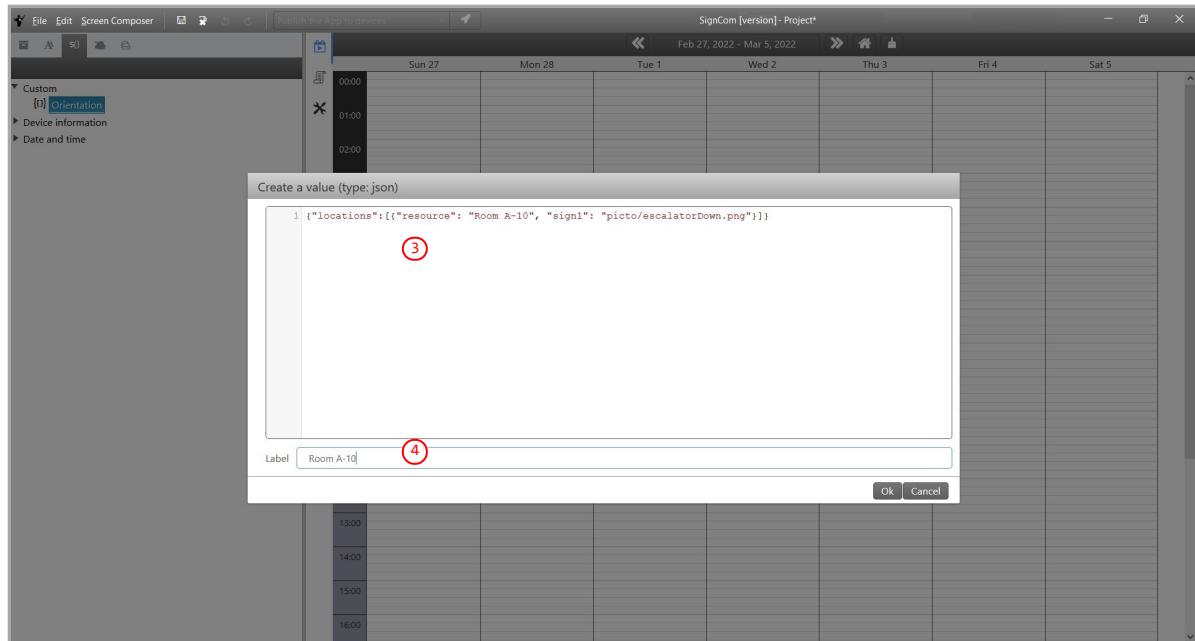
When creating a new `custom` variable of type `json` **{0}**, select the `json` type. Enter a name for your `json` variable (e.g. *Orientation*).



Add the required values for the `custom` variable of type `json` {} by pasting or typing properly the json part by respecting the `.json` syntax.

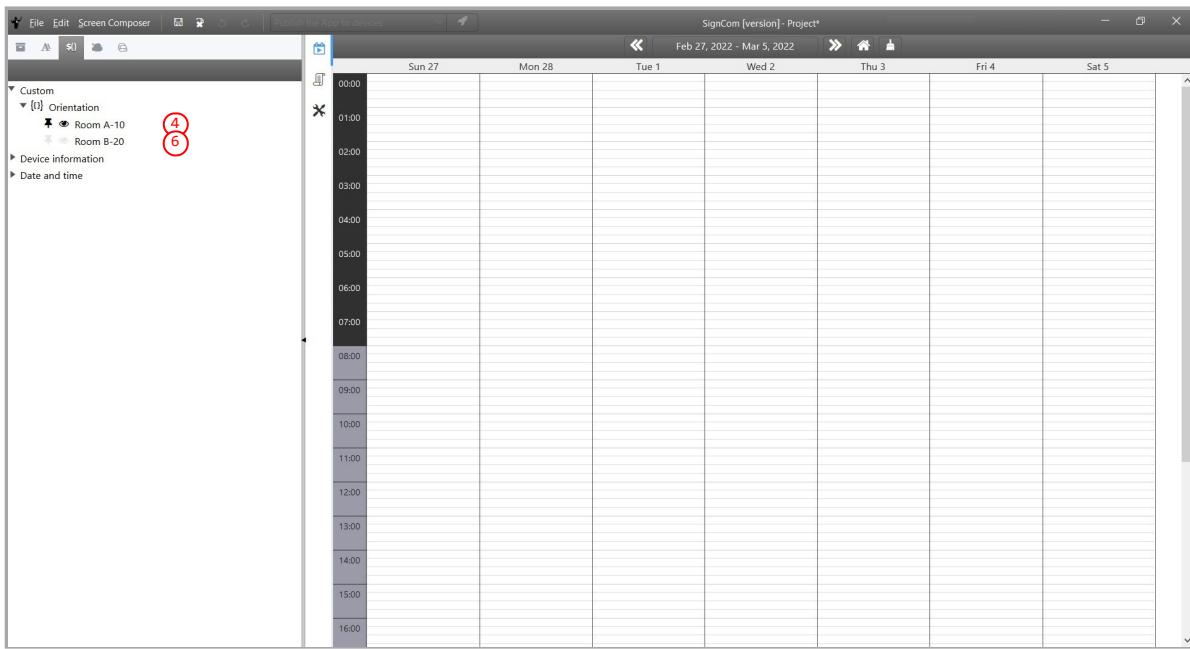


Once done, enter a label then press on the OK button.



Restart the operation to add other values for the `custom` variable of type `{0}` json.

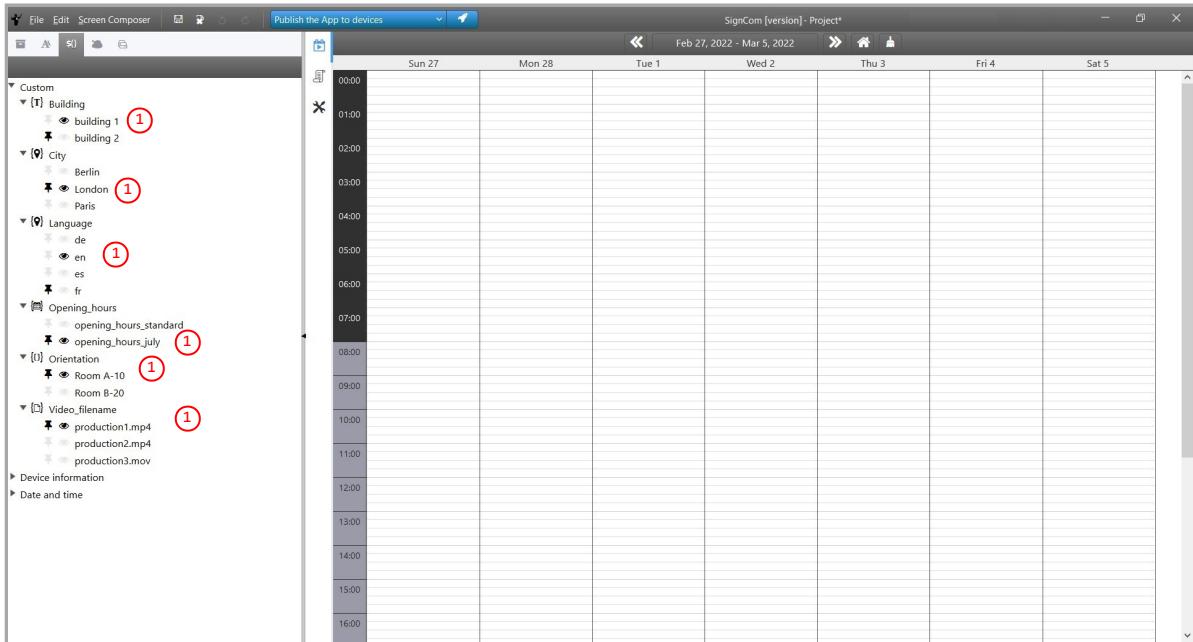
This is a summary when all the required values for the `custom` variable of type `{0}` json are created.



Default variable value for preview

When all the `custom` variables values are created, the end user has to ensure to have chosen, among all the values created for this `custom` variable, the values for preview (1) which are used when some medias, using this `custom` variable names, are previewed in the media unitary preview or in the calendar time slot preview.

An eye pictogram (1) with a black style is available in front of the values used for preview. To change the value for preview (1) for this `custom` variable, select the grey eye pictogram from another `custom` variable value. That's move automatically the eye pictogram (1) with a black style in front of the new `custom` variable value selected.



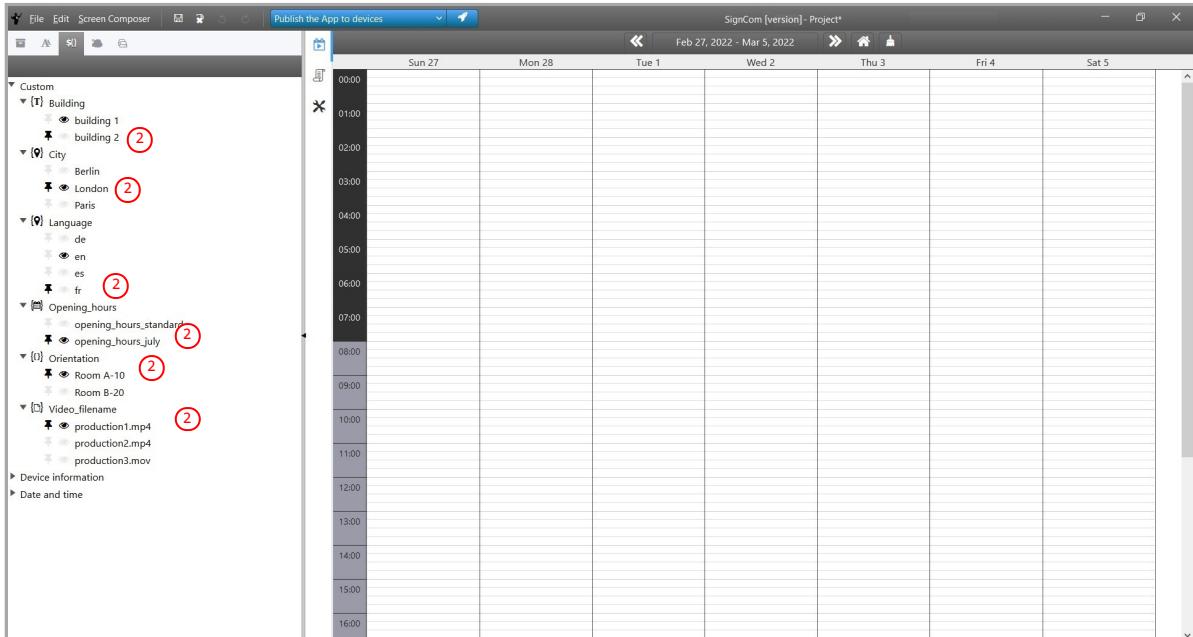
If no value is created for a `custom` variable name, the end user cannot choose a value for the preview. In this case, the value for the preview cannot be resolved with a consistent value and is by default resolved with an empty string.

Default variable value for publication

When all the `custom` variables values are created, the end user has to ensure to have chosen, among all the values created for this `custom` variable, the right default variable values (2) which are published on the device after a publication.

Once the publication is done, the medias using this `custom` variable names resolve the `custom` variable name by using the `custom` variable value set by the user in the project.

An pushpin pictogram (2) with a black style is available in front of the values used for preview. To change the value for preview (2) for this `custom` variable, select the grey pushpin pictogram (2) from another `custom` variable value. That's move automatically the pushpin pictogram (2) with a black style in front of the new `custom` variable value selected. An pushpin pictogram (2) with a black style is available in front of the values used for preview. To change the value for preview (2) for this `custom` variable, select the grey pushpin pictogram (2) from another `custom` variable value. That's move automatically the pushpin pictogram (2) with a black style in front of the new `custom` variable value selected.

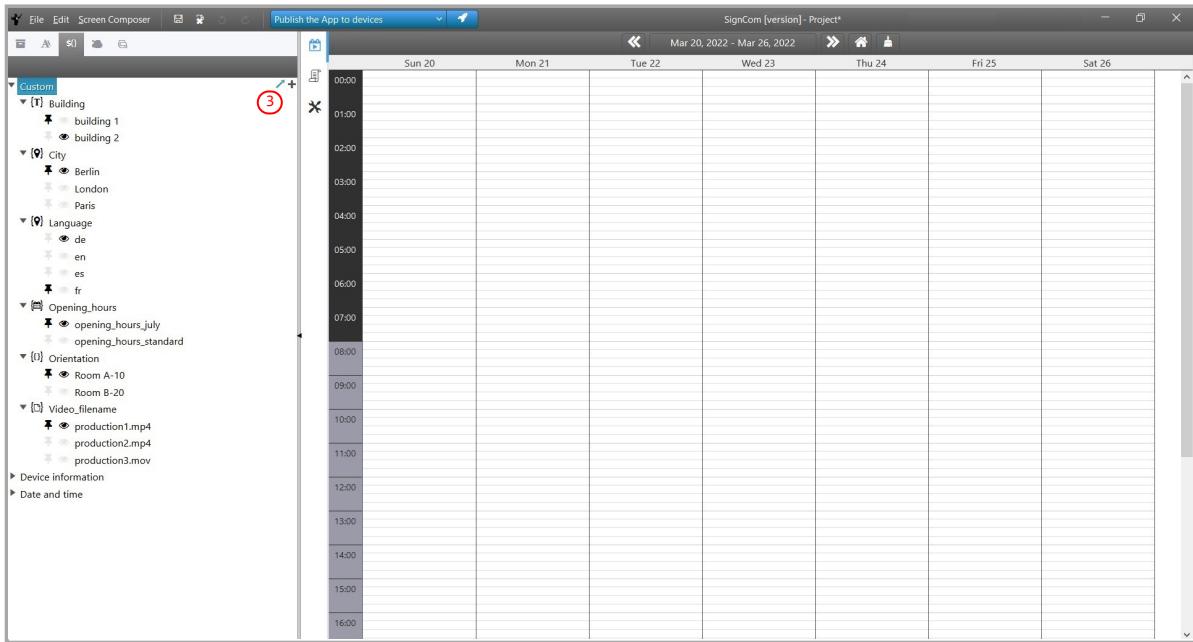


If no value is created for a `custom` variable name, the end user cannot choose a default value for the publication. In this case, no default value for this `custom` variable name can be published on the fleet of devices when the App is published.

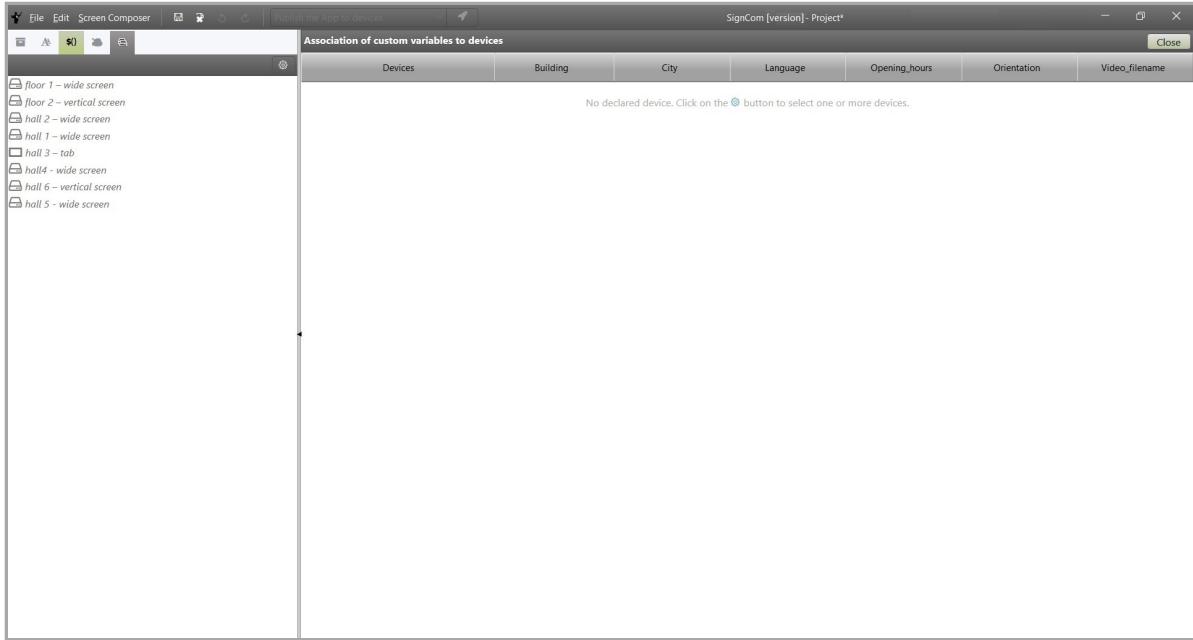
Custom variable value for publication

If the default `custom` variable value is not consistent for some specific devices, you can decide to affect to them `custom` variable values which are not the default `custom` variable values.

With your mouse, fly over the `Custom` item then click on the `Associate custom variable to device` (3) button.



To affect `custom` variable values to some devices, ensure to have properly registered devices as candidate for the publication else this message can be shown.



Drag'n drop a specific `custom` variable value (e.g `Paris` ④) to some cells ⑤ of the appropriate `custom` variable column to affect it to only some device available in the table.

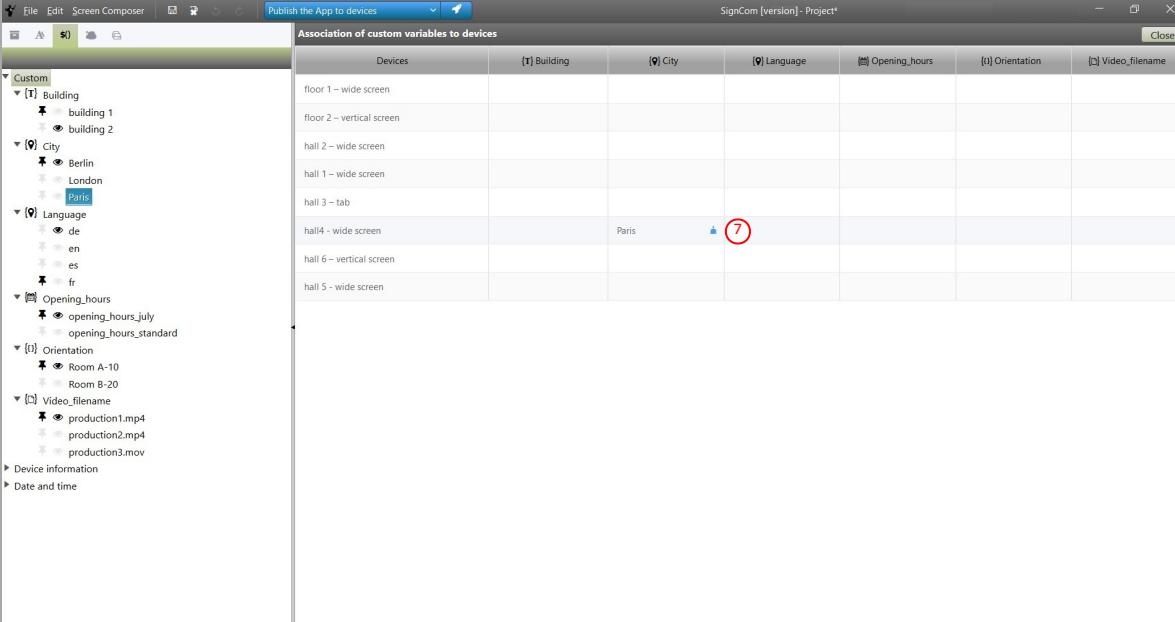
Devices	[T] Building	[Q] City	[Q] Language	[Q] Opening_hours	[I] Orientation	[C] Video_filename
floor 1 – wide screen						
floor 2 – vertical screen						
hall 2 – wide screen						
hall 1 – wide screen						
hall 3 – tab						
hall4 – wide screen		Paris				
hall 6 – vertical screen						
hall 5 – wide screen						

In this example, at the next publication:

- the value for publication `Paris` ④ for the `City` `custom` variable of type `localized` ⑤ will be published on the device with the label `hall 4 - wide screen`, meaning that the medias of the App using this variable will replace `[$custom:City]` string by `Paris` string,
- the default value `Berlin` ⑥ for publication for the `City` `custom` variable of type `localized` ⑤ will be published on all the other devices, meaning that the medias of the App using this variable will replace `[$custom:City]` string by `Berlin` string.

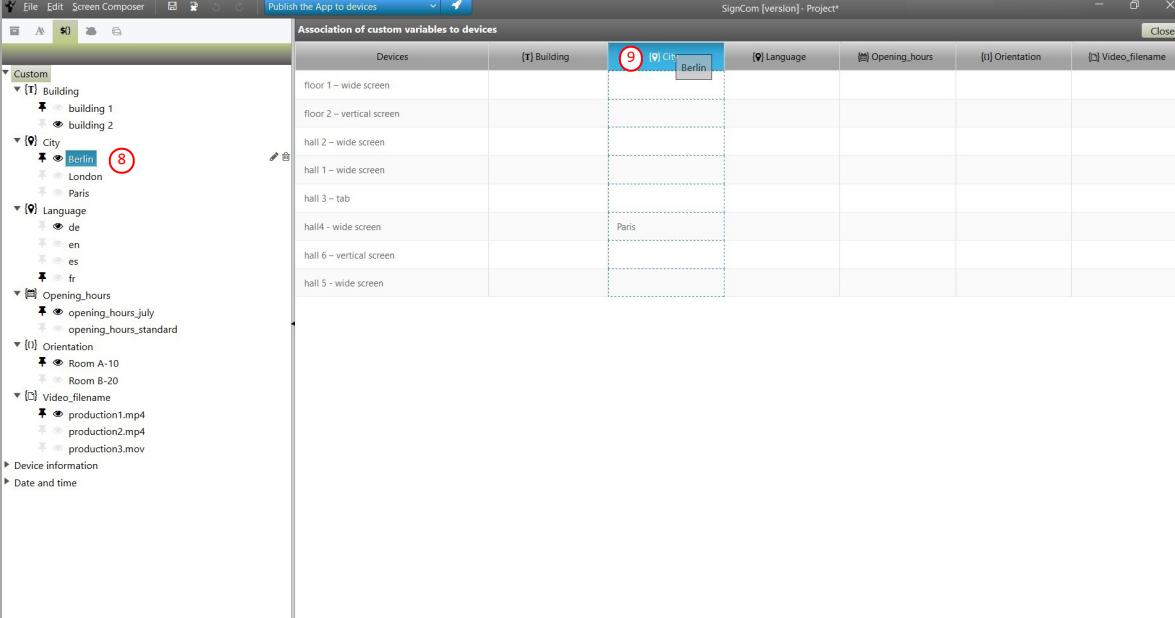
Devices	[T] Building	[Q] City	[Q] Language	[Q] Opening_hours	[I] Orientation	[C] Video_filename
floor 1 – wide screen						
floor 2 – vertical screen						
hall 2 – wide screen						
hall 1 – wide screen						
hall 3 – tab						
hall4 – wide screen		Paris				
hall 6 – vertical screen						
hall 5 – wide screen						

To delete a variable value affected to a device, fly over it with your mouse to make appear the **sweep**  button, and click on it.



Devices	{T} Building	{Q} City	{Q} Language	{M} Opening_hours	{I} Orientation	{C} Video_filename
floor 1 – wide screen						
floor 2 – vertical screen						
hall 2 – wide screen						
hall 1 – wide screen						
hall 3 – tab						
hall4 - wide screen		Paris				
hall 6 – vertical screen						
hall 5 - wide screen						

Drag'n drop a specific `custom` variable value (e.g. `Berlin` ) in the header  of the appropriate `custom` variable column to affect it to all the devices available in the table.

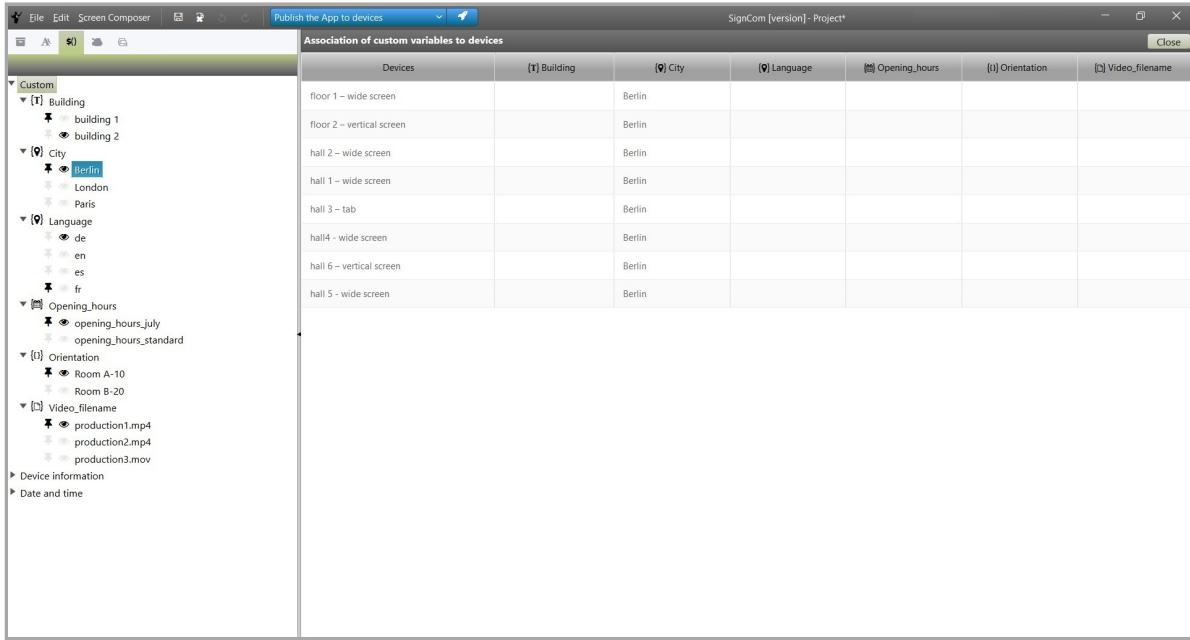


Devices	{T} Building	 {Q} City	{Q} Language	{M} Opening_hours	{I} Orientation	{C} Video_filename
floor 1 – wide screen		Berlin				
floor 2 – vertical screen						
hall 2 – wide screen						
hall 1 – wide screen						
hall 3 – tab						
hall4 - wide screen						
hall 6 – vertical screen						
hall 5 - wide screen						

In this example, at the next publication:

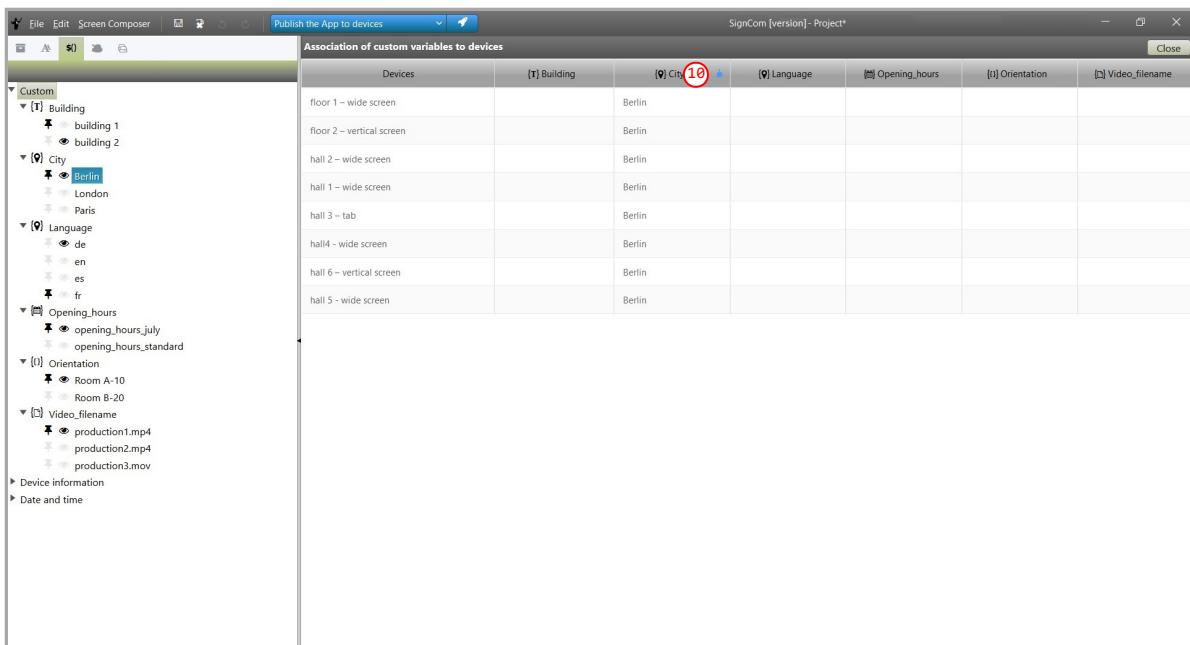
- the value for publication Berlin  for the custom localized variable name City will be published on all the other devices, meaning that the medias of the App using this variable will replace {\$custom:City} string by Berlin string.

 When all the lines of the table are filled for a given column, the default value for publication for this variable is not used.



Association of custom variables to devices						
	Devices	[T] Building	[Q] City	[Q] Language	[Q] Opening_hours	[I] Orientation
	floor 1 – wide screen		Berlin			
	floor 2 – vertical screen		Berlin			
	hall 2 – wide screen		Berlin			
	hall 1 – wide screen		Berlin			
	hall 3 – tab		Berlin			
	hall4 - wide screen		Berlin			
	hall 6 – vertical screen		Berlin			
	hall 5 – wide screen		Berlin			

To delete all the variable values affected to a device, fly over the appropriate column header with your mouse to make appear the  button, and click on it.



Association of custom variables to devices						
	Devices	[T] Building	[Q] City 	[Q] Language	[Q] Opening_hours	[I] Orientation
	floor 1 – wide screen		Berlin			
	floor 2 – vertical screen		Berlin			
	hall 2 – wide screen		Berlin			
	hall 1 – wide screen		Berlin			
	hall 3 – tab		Berlin			
	hall4 - wide screen		Berlin			
	hall 6 – vertical screen		Berlin			
	hall 5 – wide screen		Berlin			

3.4.2 Device information variable

The device information `$` variables have predefined device information variable name:

- `① field1`,
- `② field2`,
- `③ field3`,
- `④ field4`,
- `⑤ field5`,
- `⑥ hostname`,
- `⑦ mac`,
- `⑧ uuid`.

When the Device information `$` variable names are dropped in an URL or an input of a content model form, the Device information `$` variables are recognized respectively with the syntax:

- `① {$deviceInfo:field1}`,
- `② {$deviceInfo:field2}`,
- `③ {$deviceInfo:field3}`,
- `④ {$deviceInfo:field4}`,
- `⑤ {$deviceInfo:field5}`,
- `⑥ {$deviceInfo:hostname}`,
- `⑦ {$deviceInfo:mac}`,
- `⑧ {$deviceInfo:uuid}`.

They are corresponding respectively to the parameters intrinsic to each Qeedji device reported in their device `status.xml`.

This is an extract of device `status.xml` for a first device:

```
<mac>00-1c-e6-02-27-bf</mac>
<hostname>DMB400-H2</hostname>
<uuid>06c10001-0000-0000-0000-001ce60227bf</uuid>
<modelName>dmdb400</modelName>
<modelNumber>4.13.15</modelNumber>
<serialNumber>01081-00001</serialNumber>
<middleware>gekkota-4</middleware>
<field1>North-America</field1>
<field2>en</field2>
<field3>Landscape</field3>
<field4>Floor1</field4>
<field5>Building-B1</field5>
```

This is an extract of device `status.xml` for another device:

```
<mac>00-1c-e6-02-23-d3</mac>
<hostname>DMB400-F1</hostname>
<uuid>07710001-0000-0000-0000-001ce60223d3</uuid>
<modelName>dmdb400</modelName>
<modelNumber>4.13.15</modelNumber>
<serialNumber>01191-00001</serialNumber>
<middleware>gekkota-4</middleware>
<field1>Europe</field1>
<field2>de</field2>
<field3>Portrait</field3>
<field4>Floor2</field4>
<field5>Building-B2</field5>
```

☞ The `field1` to `field5` Device information `$` variable values are empty by default when the device is coming straight from factory. Refer to the device user manual to associate an intrinsic `field1` to `fields5` value to your device if required.

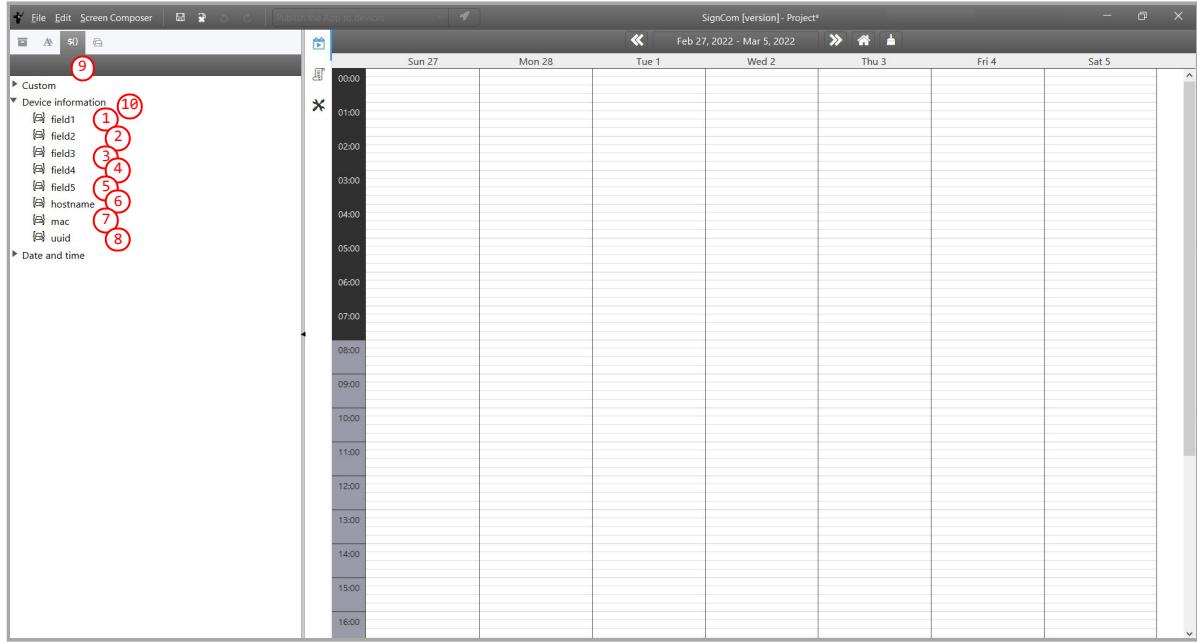
A device information `$` variable name can have one or several device information variable values.

This is an example of `device information` variable values.

device information variable name	device information variable values	label for device information variable values
① field1	<ul style="list-style-type: none"> ⑭ Africa ⑮ Antarctica ⑯ Asia ⑰ Australia ⑱ Europe ⑲ North-America ⑳ South-America 	
② field2	<ul style="list-style-type: none"> en fr de es 	
③ field3	<ul style="list-style-type: none"> Landscape Portrait 	
④ field4	<ul style="list-style-type: none"> Floor1 Floor2 	
⑤ field5	<ul style="list-style-type: none"> Building-B1 Building-B2 	
⑥ hostname	<ul style="list-style-type: none"> DMB400-F1 DMB400-H2 	
⑦ mac	<ul style="list-style-type: none"> 00-1c-e6-02-27-bf 00-1c-e6-02-23-d3 	
⑧ uuid	<ul style="list-style-type: none"> 6c10001-0000-0000-0000-001ce60227bf 0710001-0000-0000-0000-001ce60223d3 	

Variable value creation

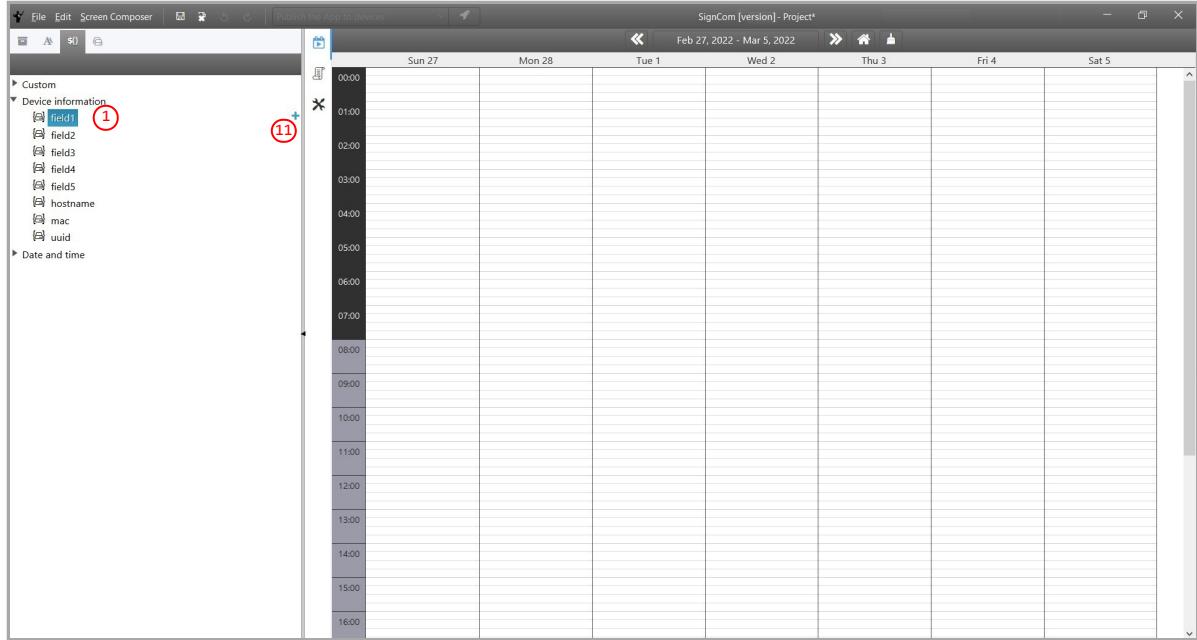
To create a Device information  variable value, click on the Variables  tab of the library and expand the Device Information  item by double-clicking on it (or by clicking on the triangle).



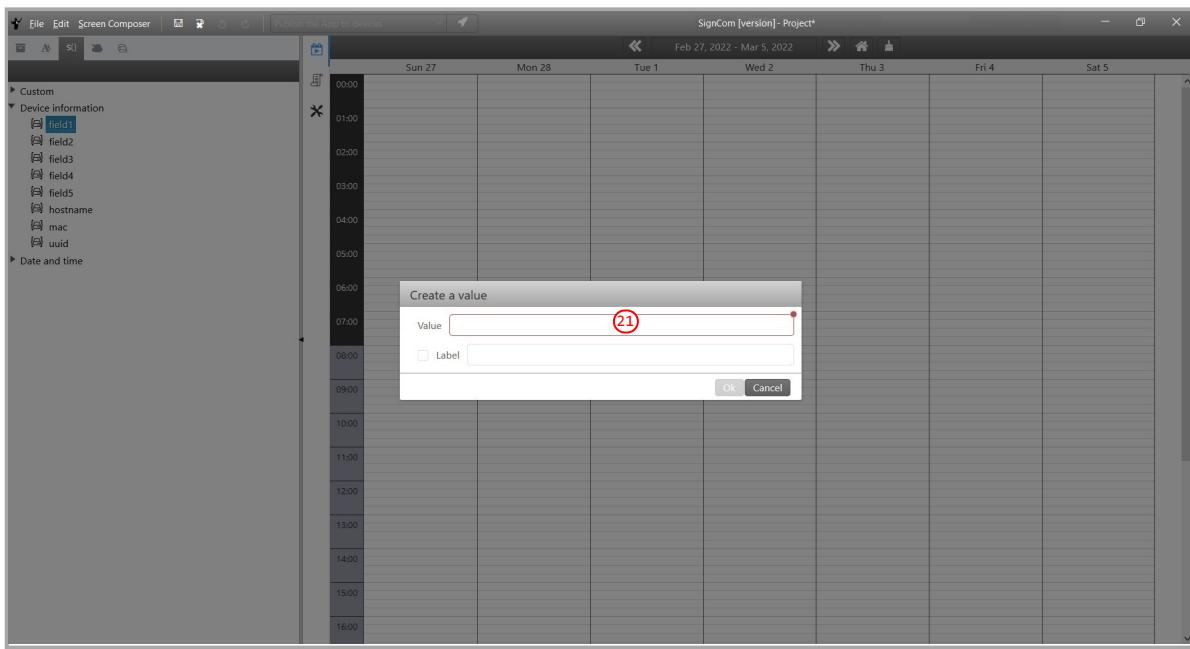
By default, there is no Device information  variable value available when Screen Composer has just been installed.

If values are existing, a black triangle appears in front of the Device information  variable. If available, click on the triangle to expand the Device information  variable name to watch the Device information  variable values available.

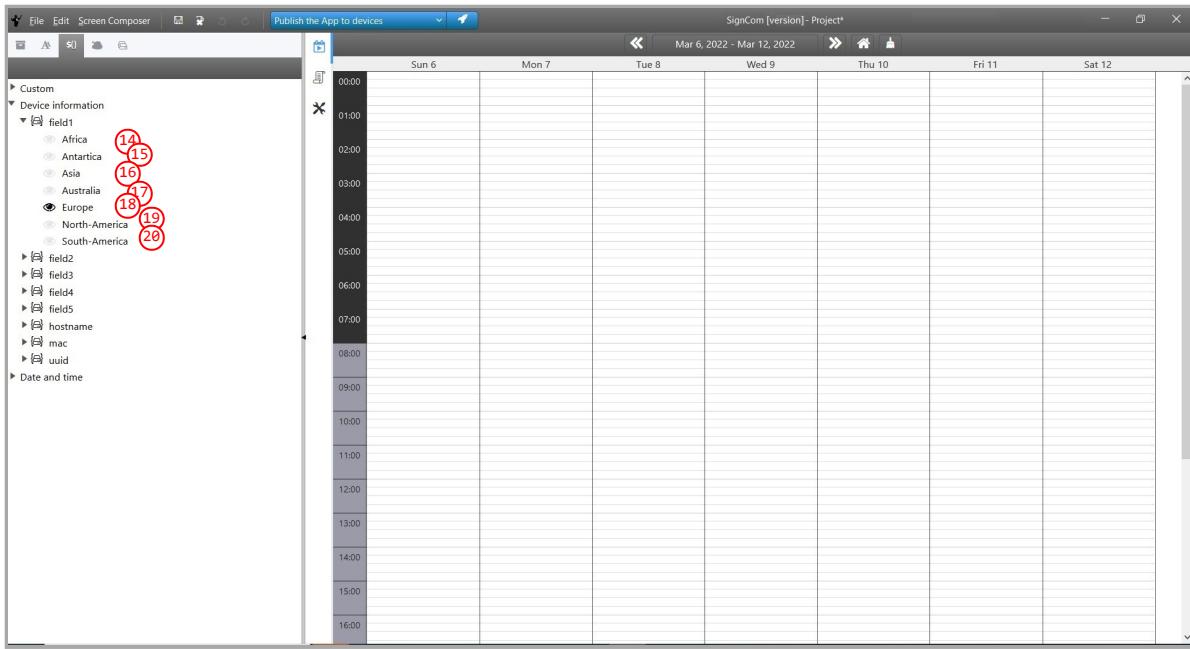
With your mouse, fly over the wished Device information  variable name to make appear the +  button.



Enter a value **(21)** for this Device information **(22)** variable name (e.g. for field1 , Europe **(18)**).



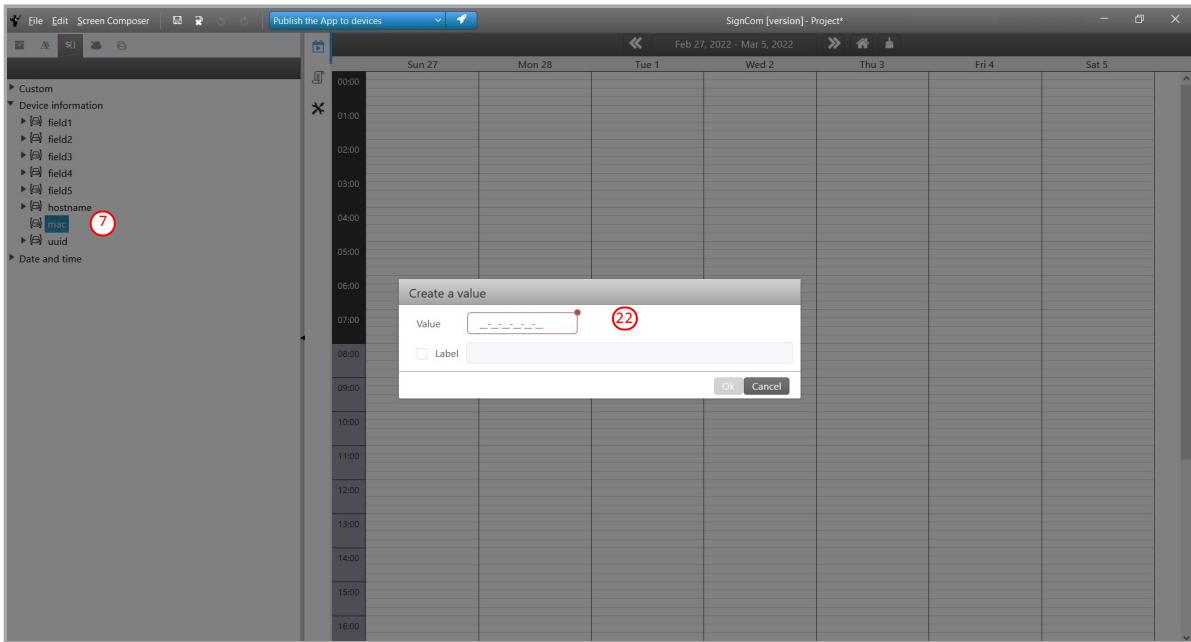
Make the operation again to add new values to this Device information **(22)** variable name (e.g. for field1 : Africa **(14)**, Antarctica **(15)**, Asia **(16)**, Australia **(17)**, North-America **(19)**, South-America **(20)**.



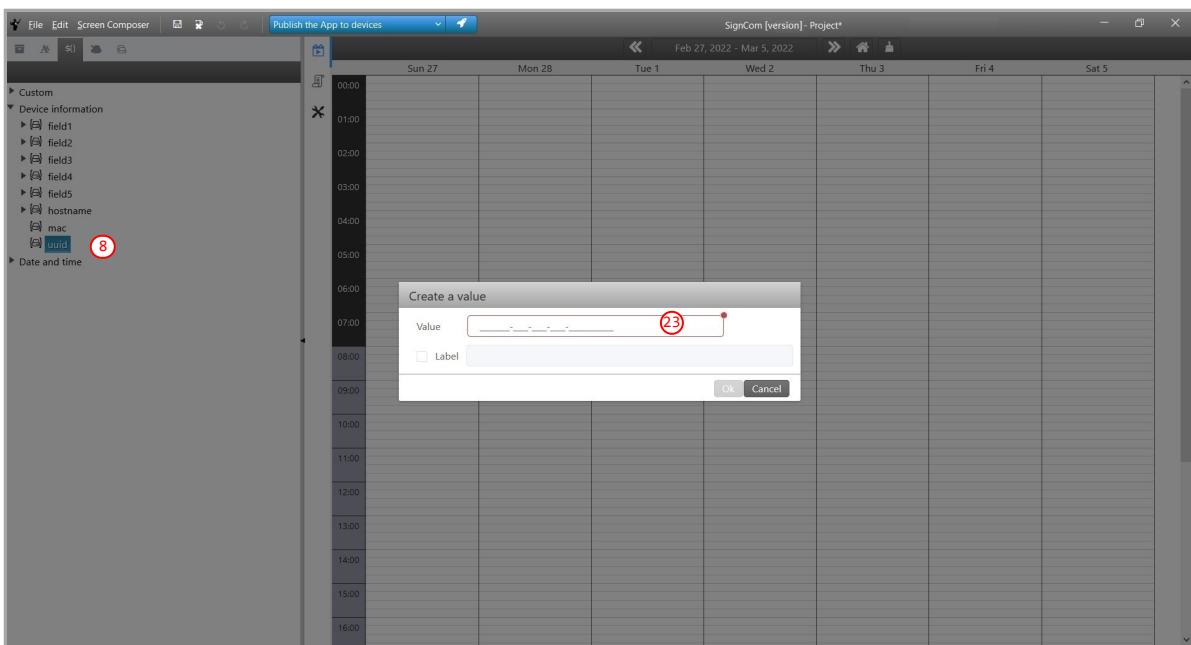
Note: The label for the Device information variable value is optional.

Make the operation again for other Device information  variables.

The popup  to enter a device information mac  variable value helps to enter a value with the appropriate format: aa-bb-cc-dd-ee-ff.



The popup  to enter a device information uuid  variable value helps to enter a value with the appropriate format: xxxxxxxx-yyyy-zzzz-kkkk-aabbccddeeff.



The Device information  variables can be used afterwards in the URL of some `URI` medias, in the `playfolder` medias or in the URL or some inputs of some content model forms:

- example of an playfolder URL input containing the `Device information` variable of type `hostname` :
`[Medias]/presentation_{$deviceInfo:hostname}.pptx`
- example of URL input containing the `Device information` variable of type `field2` : `https://www.innes.pro/{$deviceInfo:field2}/` .

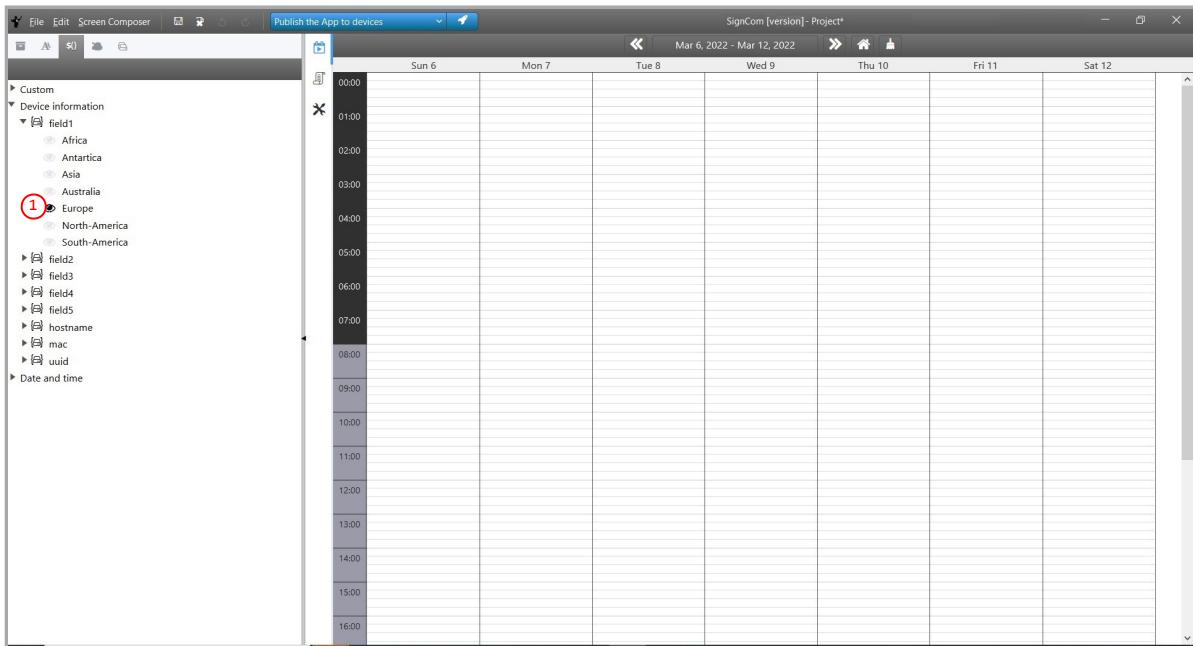
 *The values for `device information` variable of type `field1` to `field5` can contain character coming from different alphabets.*

 Publishing App having `playfolder` or `URI` using `Device information`  variable requires to add the manually the required target files or target folders in the manifest. For further information, refer to the chapter § [SignCom > Manifest](#).

Default variable value for preview

When all the `device information` variables values are created, the end user has to ensure to have chosen, among all the values created for this `device information` variable, the value for preview (1) which is used when a media, using this `device information` variable, is previewed in media unitary preview or in calendar time slot preview.

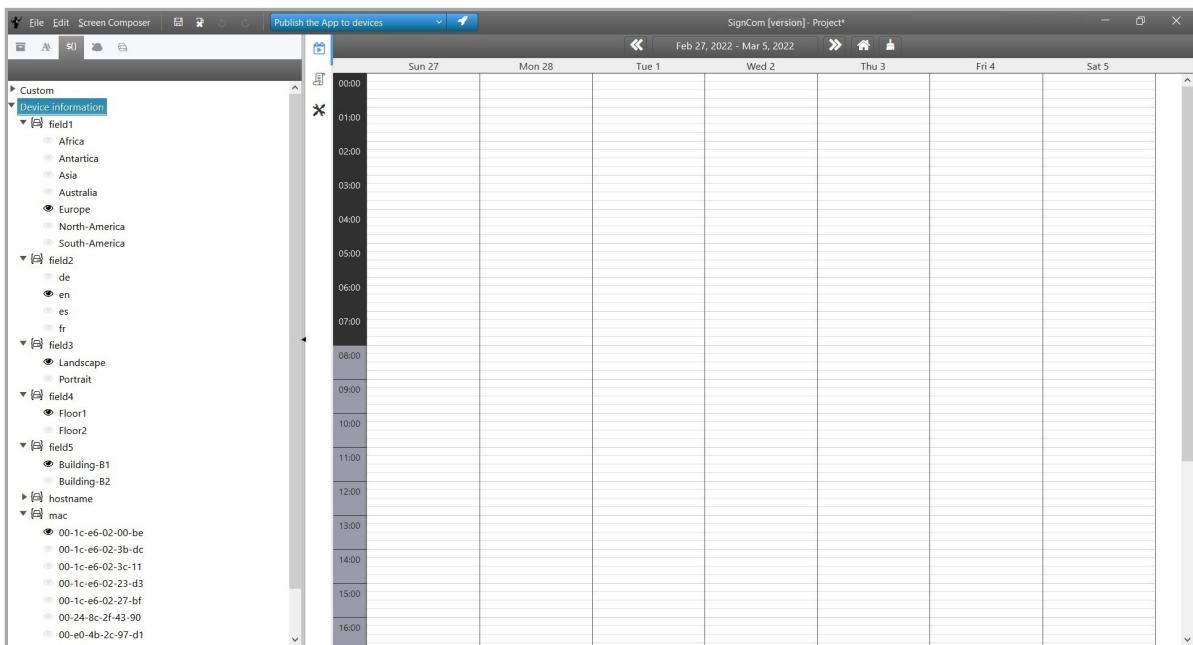
An eye (1) with a black style is available in front of the value used for preview. To change the value for preview (1) for this `device information` variable, select another `device information` variable value. That's move automatically the eye (1) in front of the new `device information` variable value selected.



If no value is created for a `device information` variable name, the end user cannot choose a value for the preview. In this case, the value for preview cannot be resolved with a consistent value and is resolved as an empty string.

The first value created for each `Device information` variable name is automatically set as the variable value for preview.

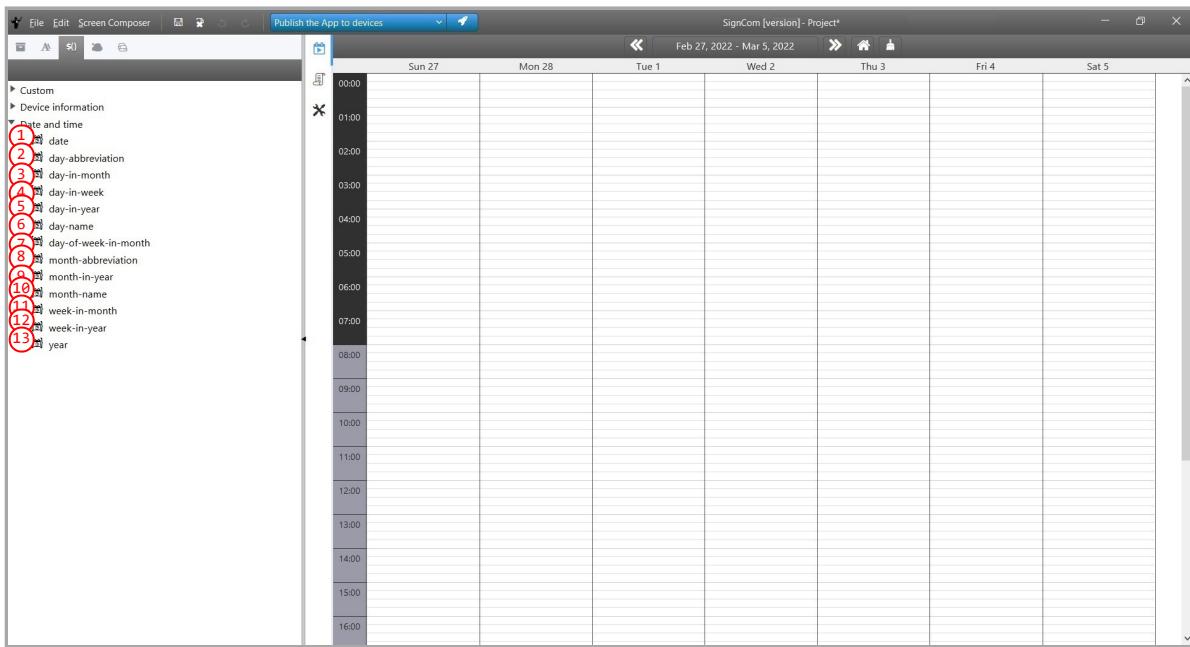
This is an example of values created in several `Device information` variable name.



3.4.3 Date and time variable

The date and time  variable name are available for any new Screen Composer project.

The date and time  variable values are depending on the day in the year and are resolved automatically by the SignCom App in preview or when published on the device.



Date and time  variable name	Format when pasted in media	Value returned in runtime	Value examples
① date	{date:date()}	current date with the format YYYY-MM-DD	2022-03-01
② day-abbreviation	{date:day-abbreviation()}	abbreviation in three characters of the day in English language, starting in upper case	Mon
③ day-in-month	{date:day-in-month()}	day number in the month from 1 to 31	5 (e.g. for the 5th of March)
④ day-in-week	{date:day-in-week()}	day number in the week from 1 for Sunday to 7 for Saturday	7 (for Saturday)
⑤ day-in-year	{date:day-in-year()}	day number in the year from 1 to 366	364
⑥ day-name	{date:day-name()}	day name in English, starting in upper case	Monday
⑦ day-of-week-in-month	{date:day-of-week-in-month()}	the number of occurrence of the current week day since the start of the month from 1 to 5	4
⑧ month-abbreviation	{date:month-abbreviation()}	abbreviation in three characters, of the current year month, in English language, starting in upper case	Nov (for the November month)
⑨ month-in-year	{date:month-in-year()}	number of the current month in the year from 1 to 12	10 (for the October month)
⑩ month-name	{date:month-name()}	current year month, in English language, starting in upper case	October
⑪ week-in-month	{date:week-in-month()}	number of the entire week or partial week elapsed since the start of the month from 1 to 6	2
⑫ week-in-year	{date:week-in-year()}	year week number from 1 to 53	52
⑬ year	{date:year()}	current year with format YYYY	2019

 Publishing App having Playfolder or URI (shorcut to files) medias using custom Date and time variable requires to add the manually the required target files or target folders in the manifest. For further information, refer to the chapter § [SignCom > Manifest](#).

3.5 Services accounts

Thanks to the services account , the SignCom App allows to create:

- some playfolders medias playing:
 - medias located on a shared directory of your local network requiring an authentication,
 - medias located on a Web directory requiring an authentication,
- some URI of Web page hosted on simple web server requiring an authentication.

The playfolder requires:

- either a CIFS services account supporting these services: - Files/folders (smb://),
- or Microsoft 365 services account supporting these services: - Files/folders (OneDrive), - Identity, - Contact, - Calendar, - Email.

The URI requires:

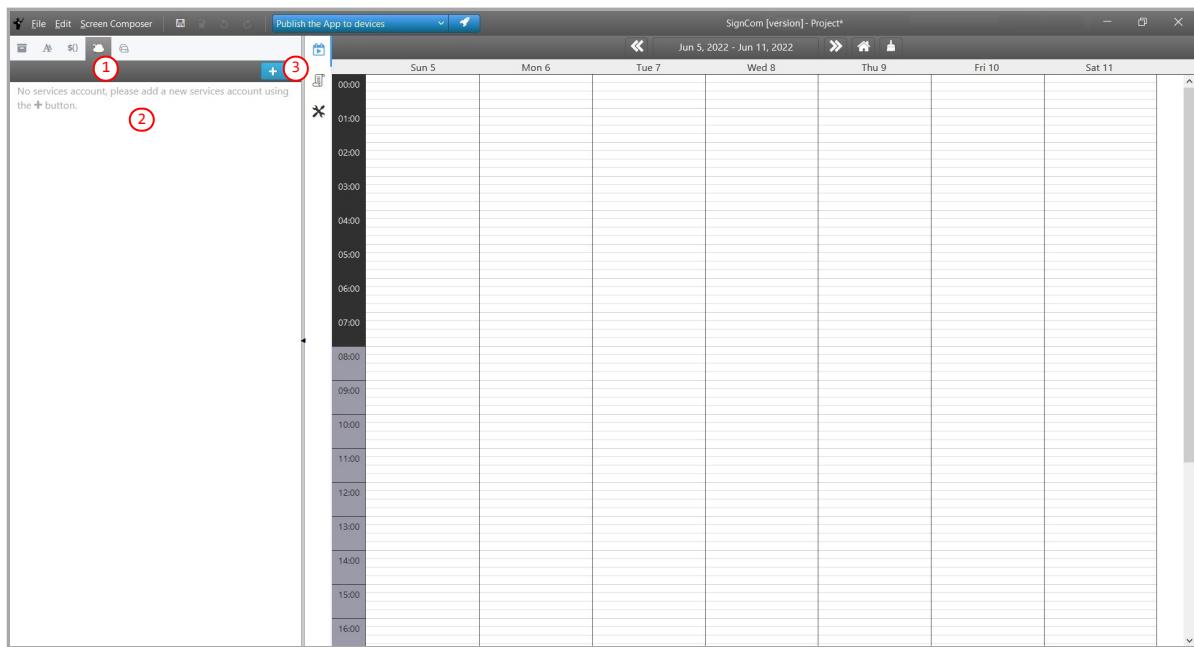
- a simple Web server services account supporting these services: - Web sites (http:// , https://).

Info Using CIFS services account may require that your local network and your computer sharing files and directories are supporting properly the protocol Microsoft SMB/CIFS 2.0.

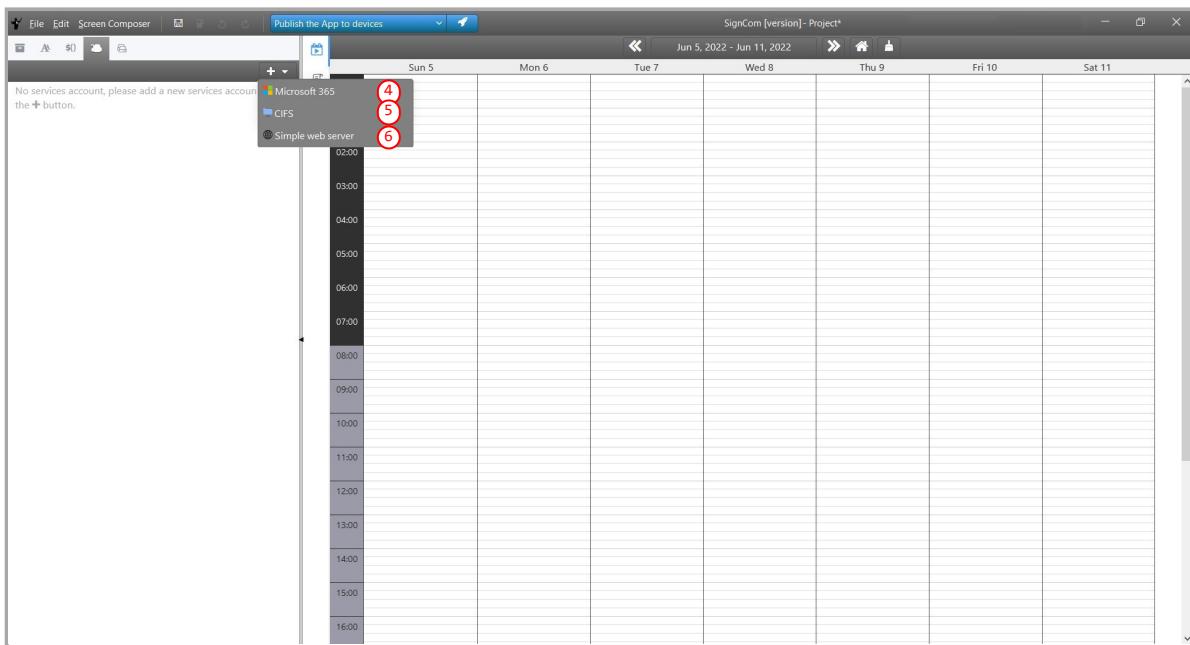
Click on the  Services accounts (1) tab.

Info By default, there is no services account (2) created.

To create a services account (3), click on the  Add a services account button.



Either click on the Microsoft 365 (4) to create a services account so that your playfolder can access to the directories of your Microsoft 365 (OneDrive); or click on the CIFS (5) to create a services account so that your playfolder can access to a shared directory of your local network (CIFS over Microsoft SMB); or select Simple web server (6) to create a services account so that your URI to play the Web page content.



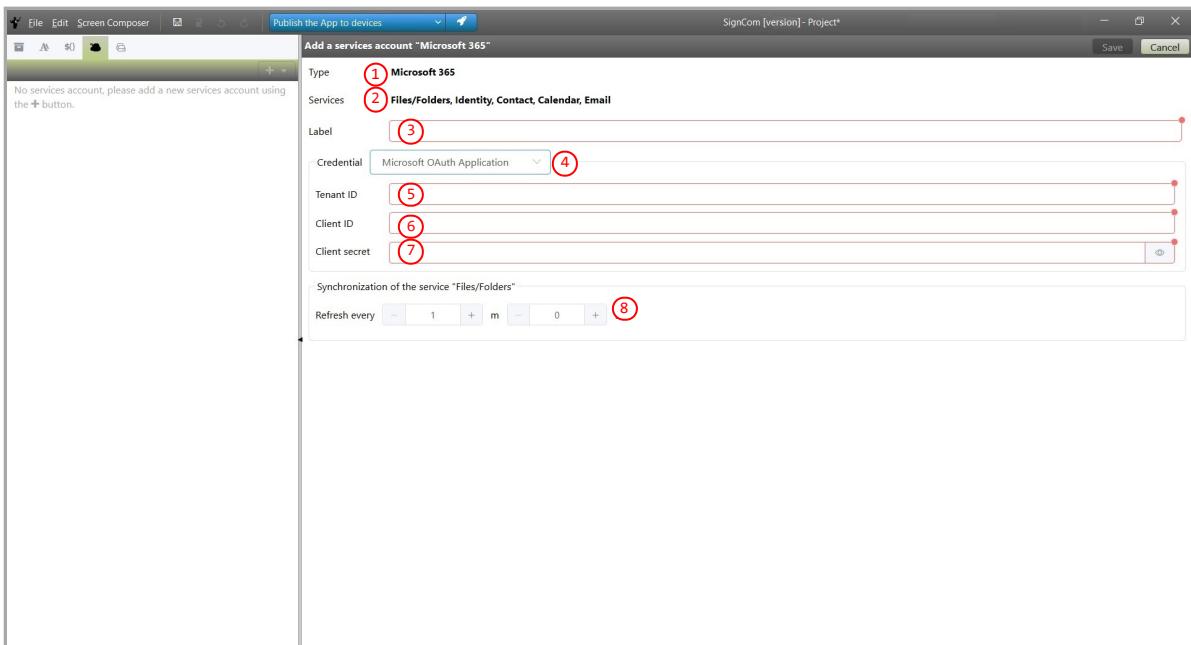
Microsoft 365 services accounts



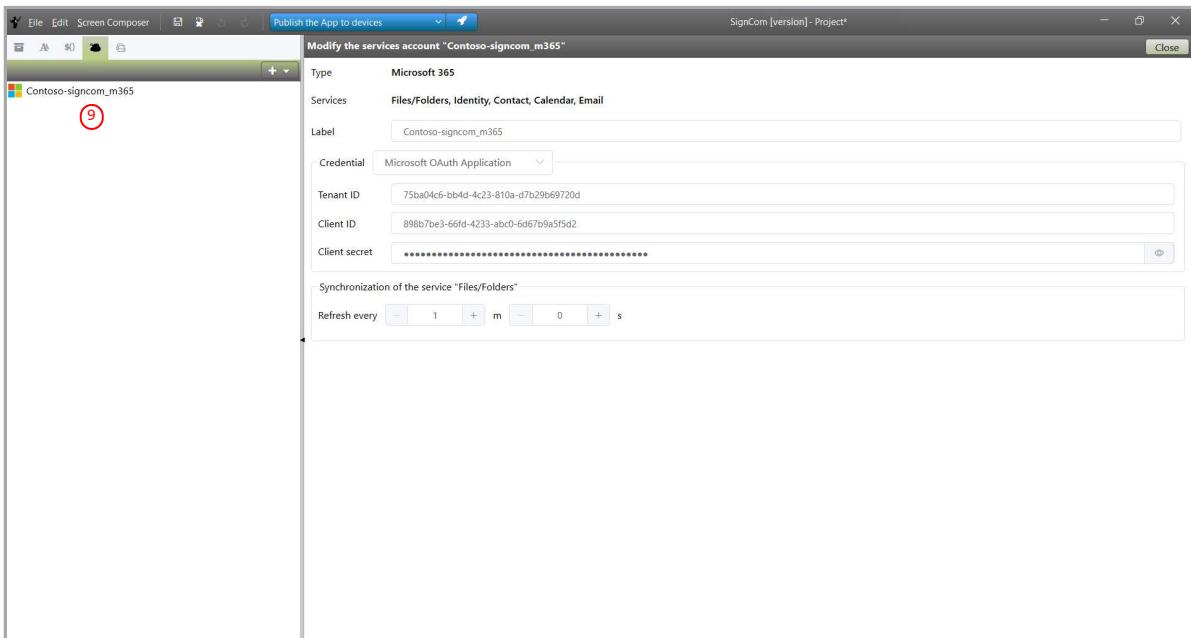
These are the different fields to create a Microsoft 365 services account :

- **① Type** : shows the services account type (e.g. Microsoft 365),
- **② Services** : shows the services supported by this services account type,
- **③ Label** : label of the services account (service account that can be selected when editing a playfolder),
- **④ Credential** : choose the credential format (e.g. Microsoft OAuth Application),
- **⑤ Tenant ID** : tenant ID of your Azure AD application ¹ to access to the Microsoft 365 shared folders of the company,
- **⑥ Client ID** : client ID of your Azure AD application ¹ to access to the Microsoft 365 shared folders of the company,
- **⑦ Client secret** : client secret of your Azure AD application ¹ to access to the Microsoft 365 shared folders of the company,
- **⑧ Refresh every** : folder content synchronization period. Every one minute by default.

¹ Refer to the chapter § [Appendix: Azure AD Application Powershell module](#) to create your Azure AD playfolder application having appropriate rights to access to the Microsoft 365 directories.



This is an example of Microsoft 365 services account (e.g. Contoso-signcom_m365). The new services account **⑨** (e.g. Contoso-signcom_m365) appears on the left side.

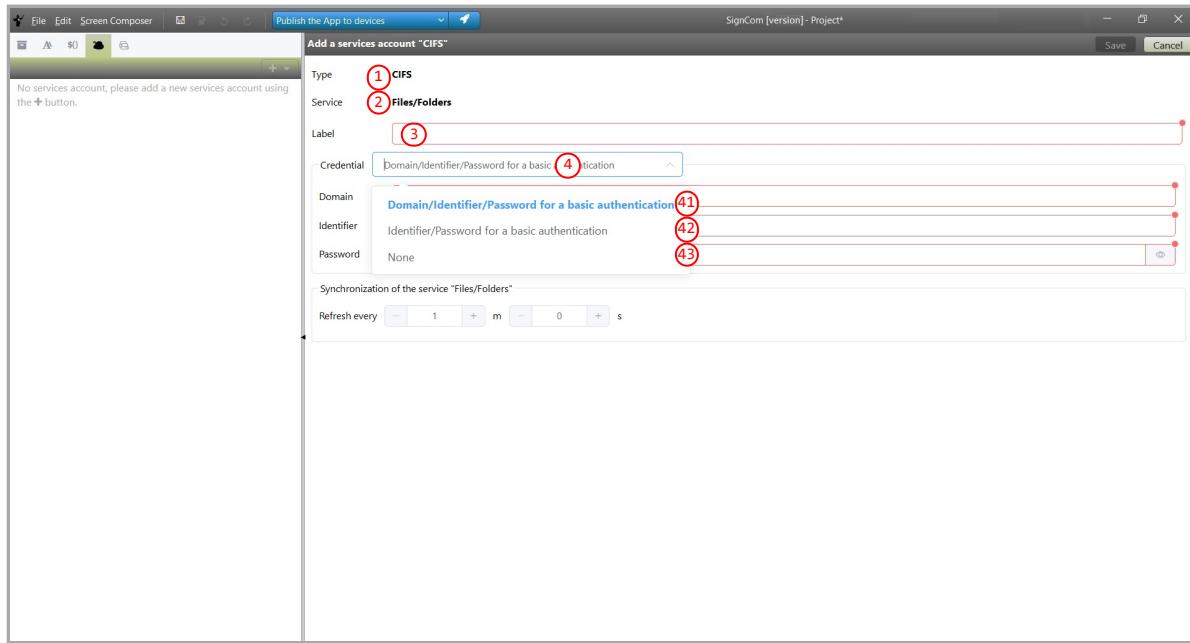


CIFS services accounts



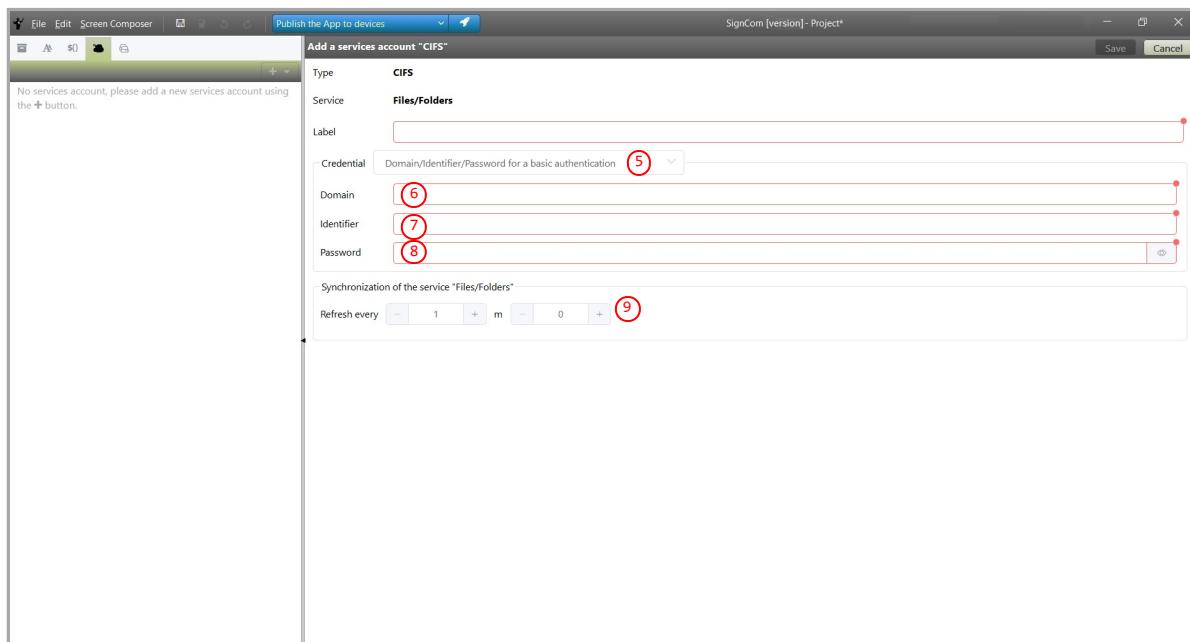
These are the different fields to create a CIFS services account :

- ① Type : shows the services account type (CIFS),
- ② Services : shows the services supported by this services account type:
 - Files/Folders,
- ③ Label : label of the services account (that can be selected for example when editing a playfolder),
- ④ Credential : choose the credential format among those supported:
 - ④1 Domain/Identifier/Password for a basic authentication,
 - ④2 Identifier/Password for a basic authentication,
 - ④3 None.



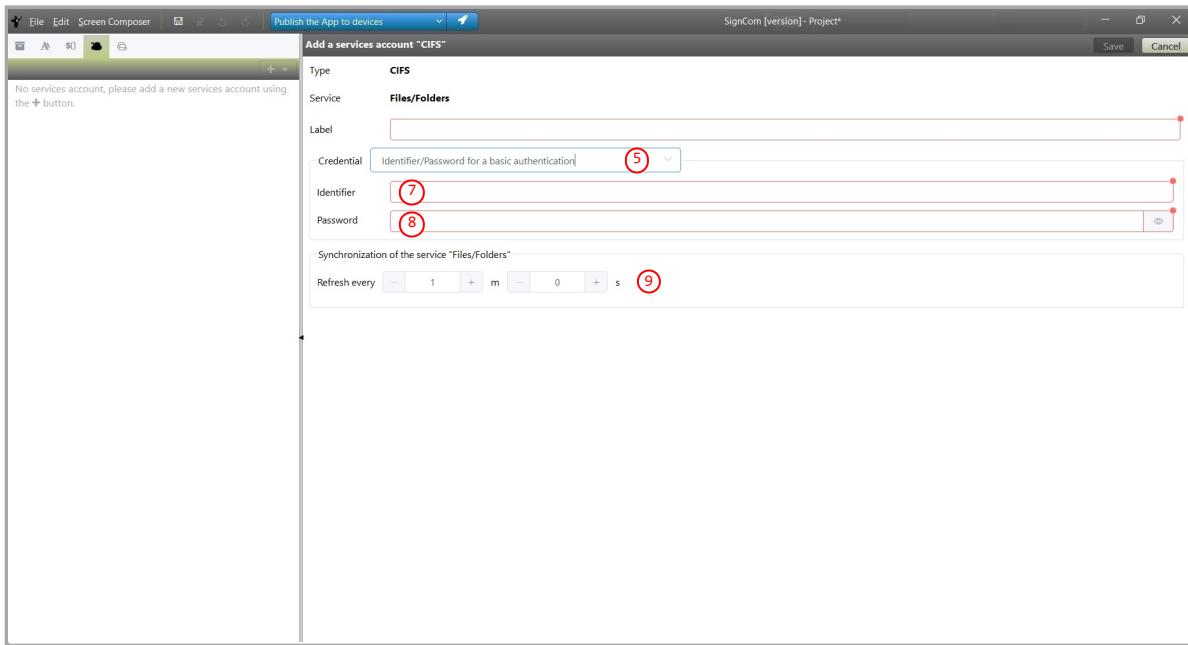
Then depending on the chosen credential format, enter the credential parameters values:

- for *Domain/Identifier/Password for a basic authentication* ⑤
 - network Domain ⑥,
 - Identifier ⑦,
 - Password ⑧.

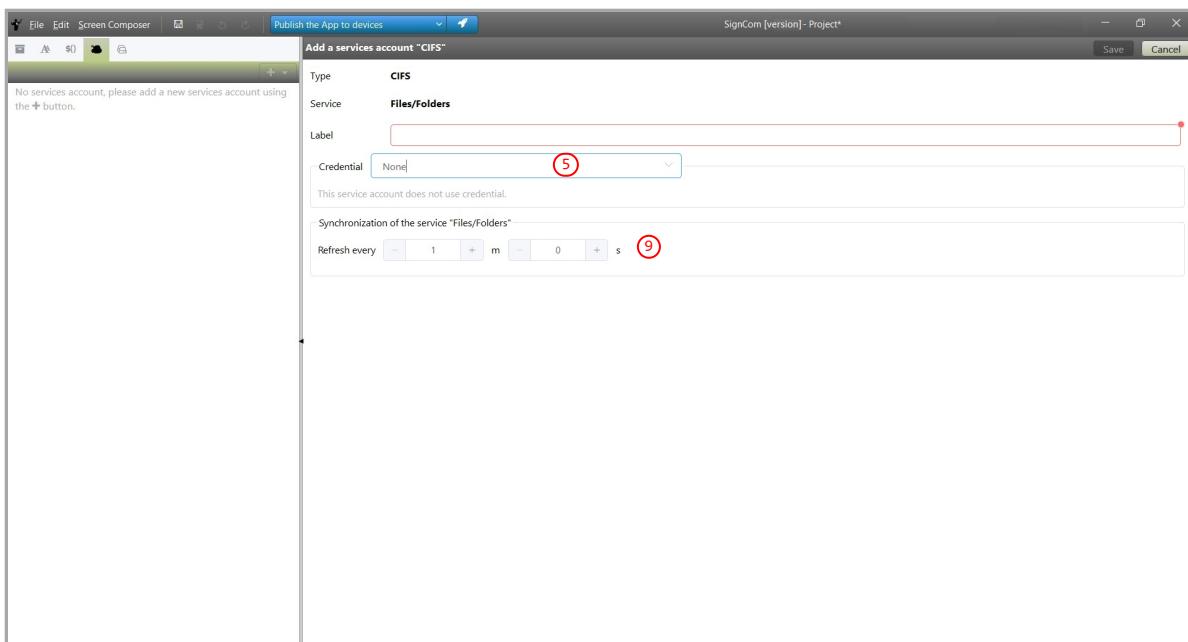


- for the *Identifier/Password* for a basic authentication (5):

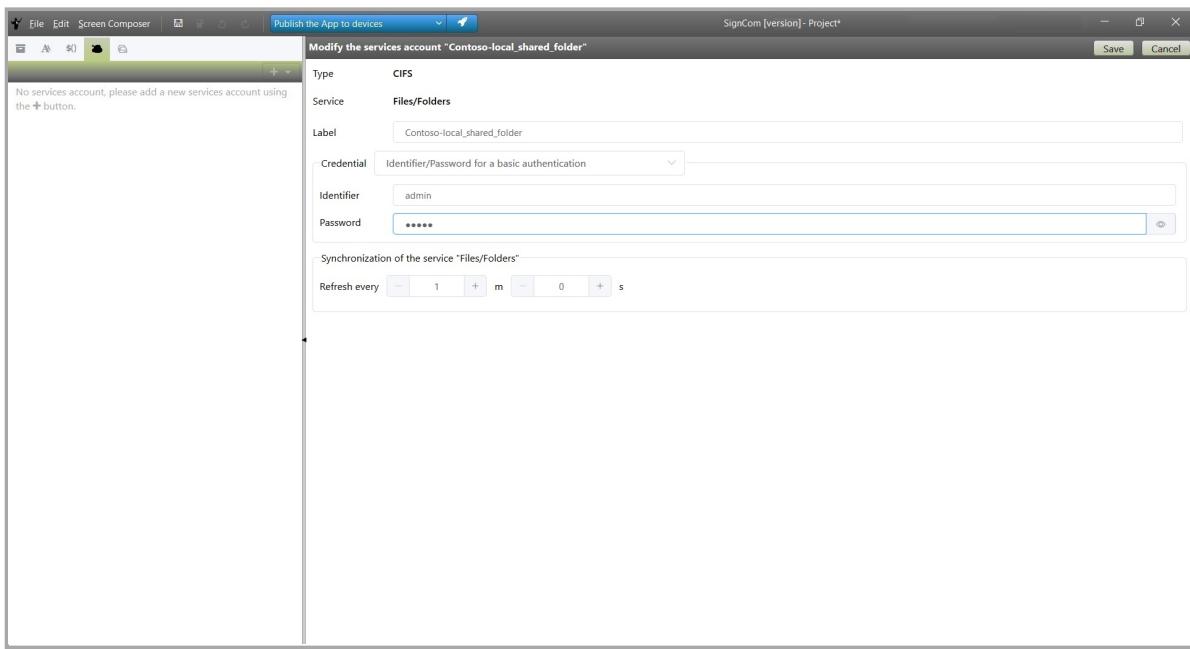
- Identifier (7),
- Password (8).



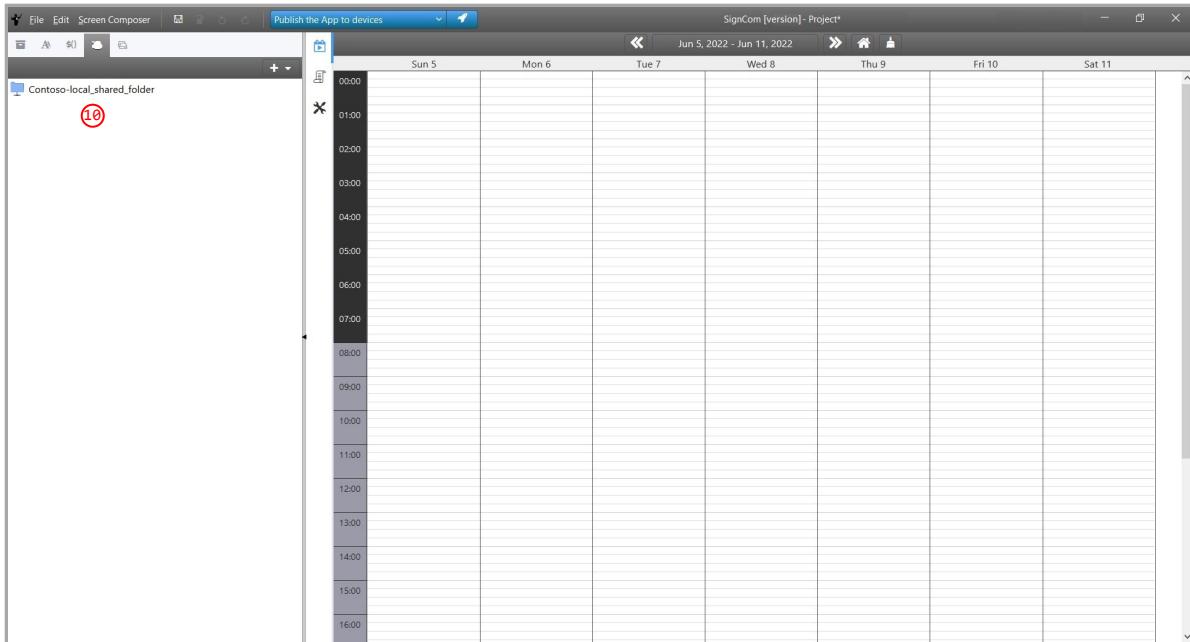
- for the *none* (5), no login credential is required.



This is an example of CIFS services account (e.g. Contoso-local_shared_folder).



The new services account (e.g. Contoso-local_shared_folder **(10)**) appears on the left side.

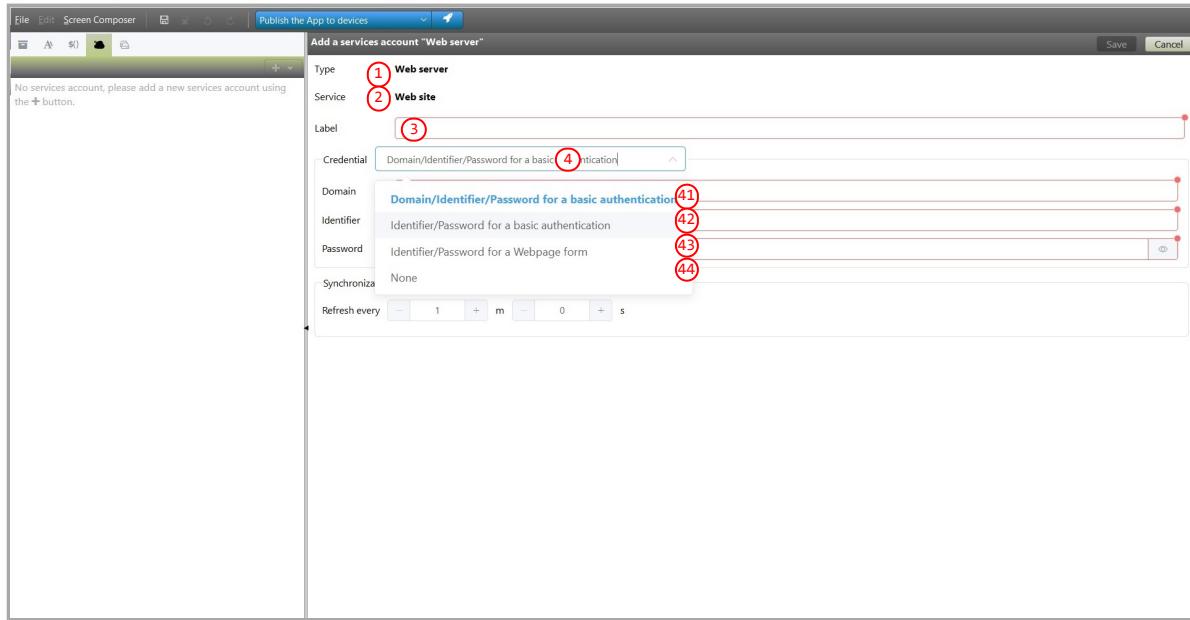


Simple web server



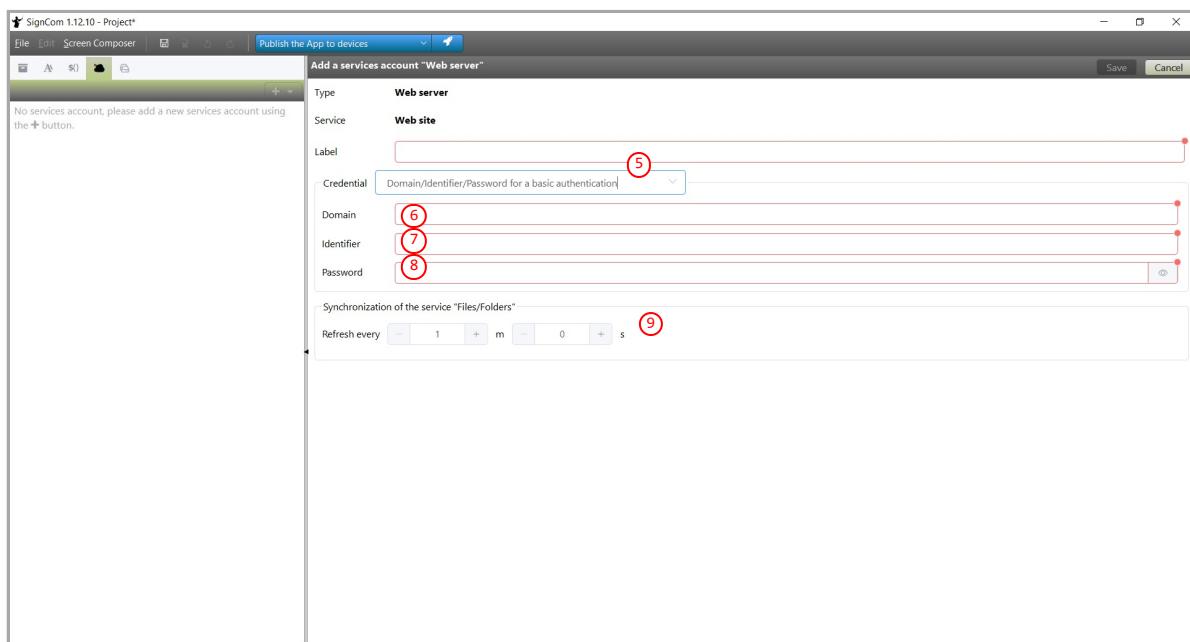
These are the different fields to create a `simple web server` services account :

- **①** Type : shows the services account type (*Simple Web server*),
- **②** Services : shows the services supported by this services account type:
 - *Web Site*,
- **③** Label : label of the services account (that can be selected when editing a URI),
- **④** Credential : choose the credential format among those supported:
 - **④1** *Domain/Identifier/Password for a basic authentication*,
 - **④2** *Identifier/Password for a basic authentication*,
 - **④3** *Identifier/Password for a Webpage form*,
 - **④4** *None*.



Then, depending on the credential type, enter the credential parameters values:

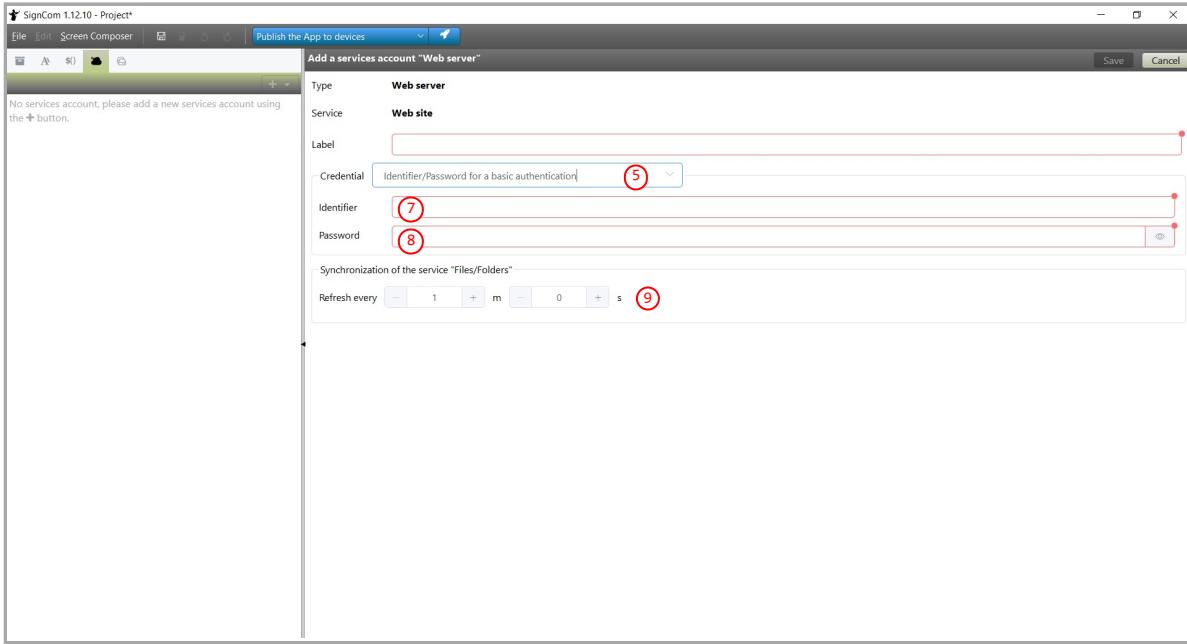
- *Domain/Identifier/Password for a basic authentication* **⑤**
 - Domain **⑥**,
 - Identifier **⑦**,
 - Password **⑧**.



- **Identifier/Password for a basic authentication** (5):

- Identifier (7),
- Password (8).

☞ Choose this credential mode when needing to fill a credential popup above the Webpage requiring a basic authentication.

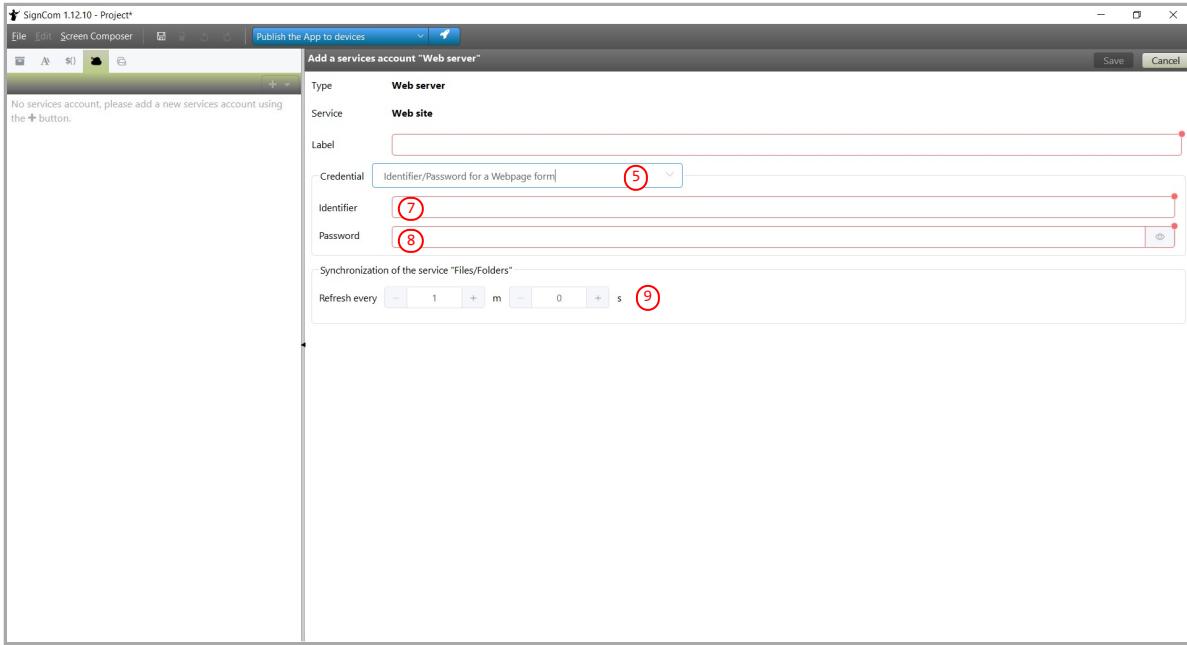


- **Identifier/Password for a Webpage form** (5):

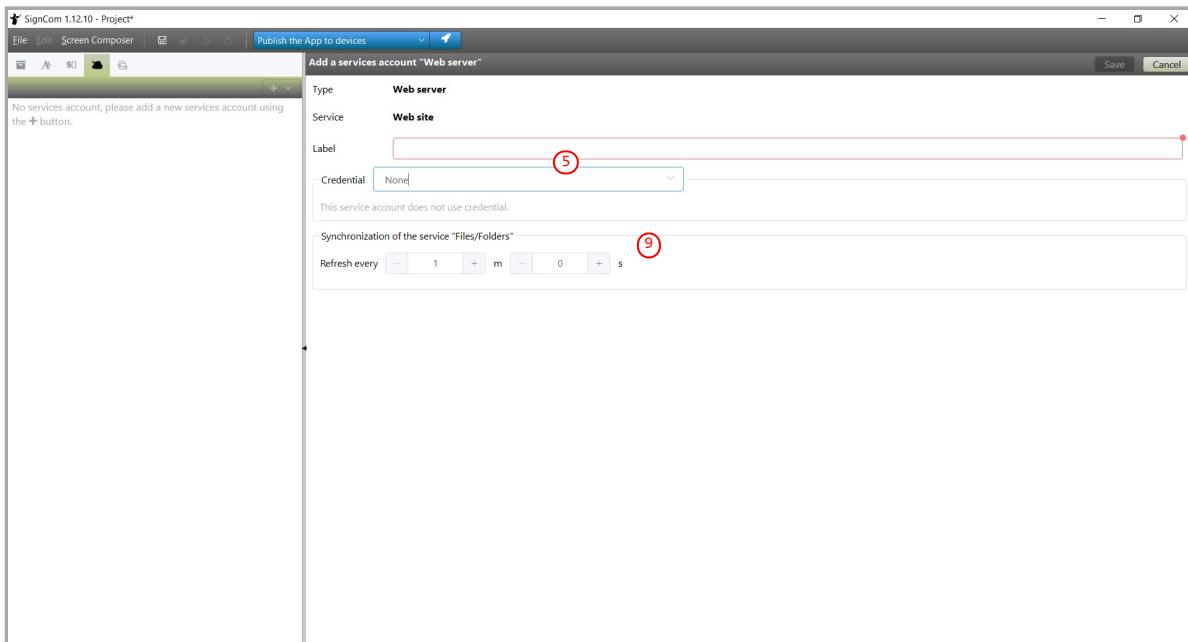
- Identifier (7),
- Password (8).

☞ Choose this credential mode when the identifier/password values for credentials must be filled inside inputs inside the Web page.

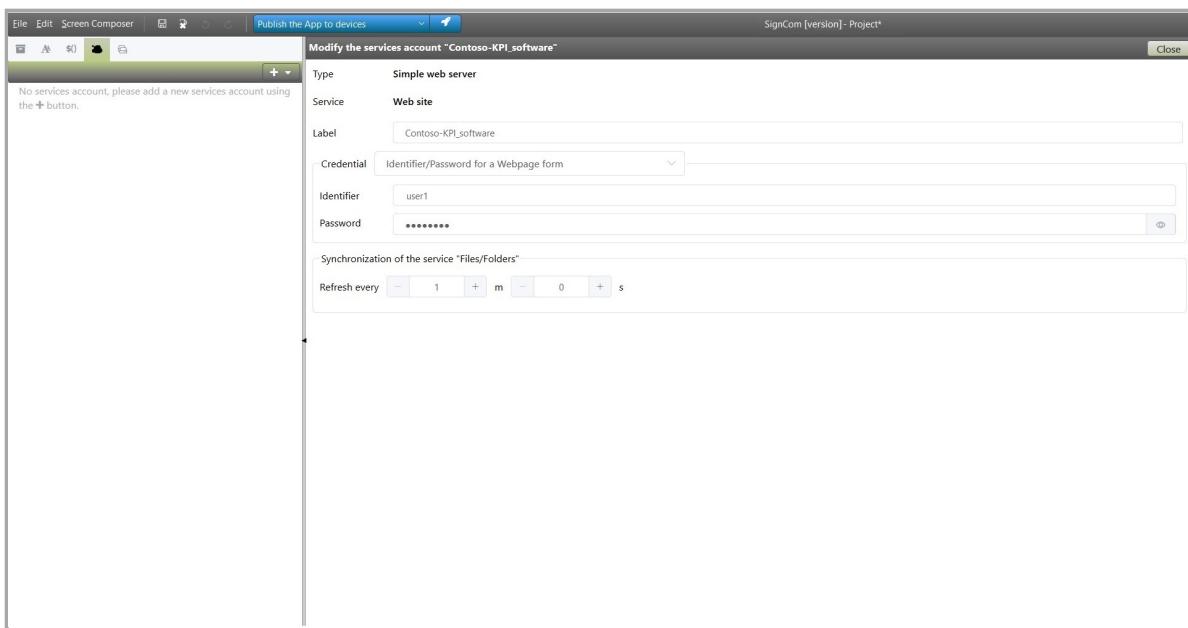
☞ For further information, refer to the chapter § [Appendix: supported login and password inputs properties and buttons properties for a Webpage form](#).



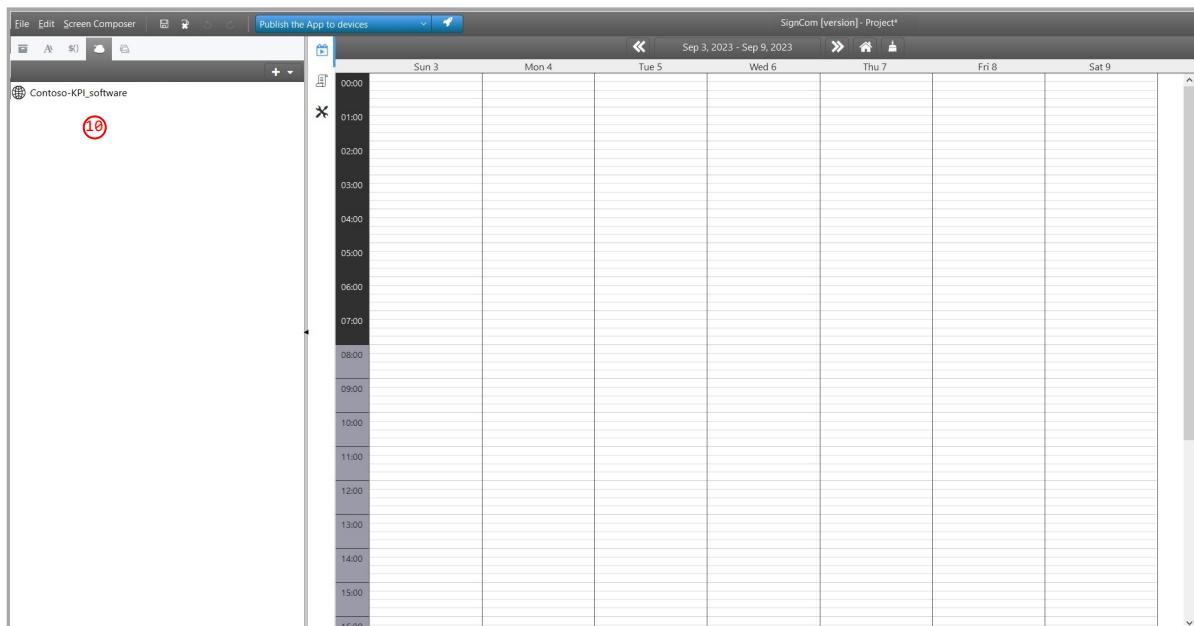
- *none* ⁽⁵⁾: no login credential parameter value is required.



This is an example of *Simple web server* services account (e.g. Contoso-KPI_software).



The new services account (e.g. Contoso-KPI_software ) appears on the left side.

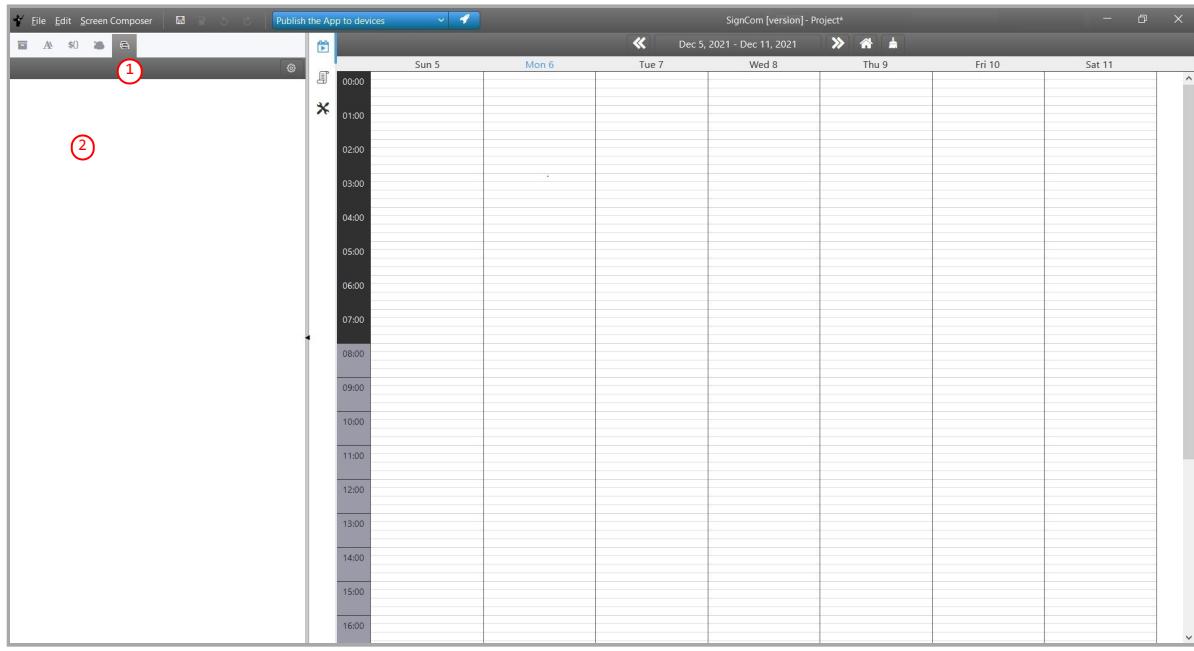


 Once one or several Simple web server services accounts have been created in the Screen Composer version 4.12.10 (or above), returning to a Screen Composer version below 4.12.10 will imply an error when publishing an App showing: n is undefined. To work around, remove all the Simple web server services accounts and publish again.

3.6 Devices

On the left side, click on the **Devices**  tab.

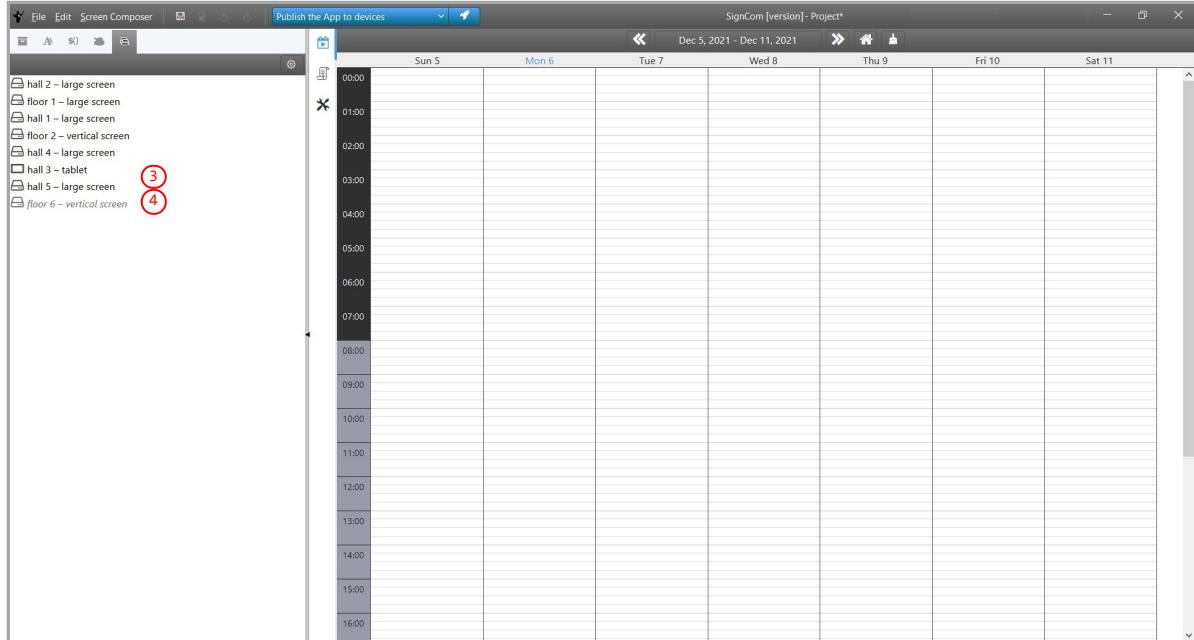
When no device is added to your project, no device is visible in the **Devices**  tab.



Click on the **Project settings > Devices**  button, short cut to the **Devices** item of the **File** menu of the **Project settings**, to add and manage the devices of your project. For further information, refer to the chapter § [File > Project settings > Devices](#).

Once the devices are added to your project, they are visible in the  **Devices** tab of your project repository.

The devices are shown in a black style  color when they are checked as applicant for publication. Else they are shown in a italic/grey style .



3.7 SignCom > Visual channel

The `Visual channel` part supports four major features:

- a view of the `channel programmation` with a timeline allowing to have an overview of the `calendar time slots` programmed week by week. For further information, refer to the chapter § [Visual channel programmation](#),
- a window allowing to modify modifying, among other things, the recurrence and exceptions of the `calendar time slot`. For more information, please refer to the chapter § [Edit the calendar time slot planning](#),
- a spatial view of the selected `calendar time slot` allowing to view, display grid per `display grid`, the whole contents prgorammed in it. For further information, refer to the chapter § [Edit the media sequence of the calendar time slot](#),
- a `calendar time slot` preview allowing to previsualize, at a defined date & time, the global content of a specific `calendar time slot` with its grid transitions and transistion between medias. For further information, , refer to the chapter § [Preview of the calendar time slot](#).

Visual channel programmation

Click on the `visual channel`  (1) button to view the `channel programmation` week by week.

The `channel programmation` is the list of all the `calendar time slots` programmed for this project for:

- all the weeks in the past,
- the current week and,
- all the weeks in the future.

A `calendar time slot` is an event of the `channel programmation` having:

- a start time and end time,
- a start date,
 - without recurring day,
 - with recurring days and with no end date,
 - with recurring days and with an end date.

The `media sequence` is the list of the medias and `display grids` inserted in a `calendar time slot` in all the `display regions` with a specific order.

The `scene` is the group of medias played and driven by a specific `display grid`. A `display grid` transition leads automatically to a scene transition.

When a project has just been created, the `channel programmation` is empty.

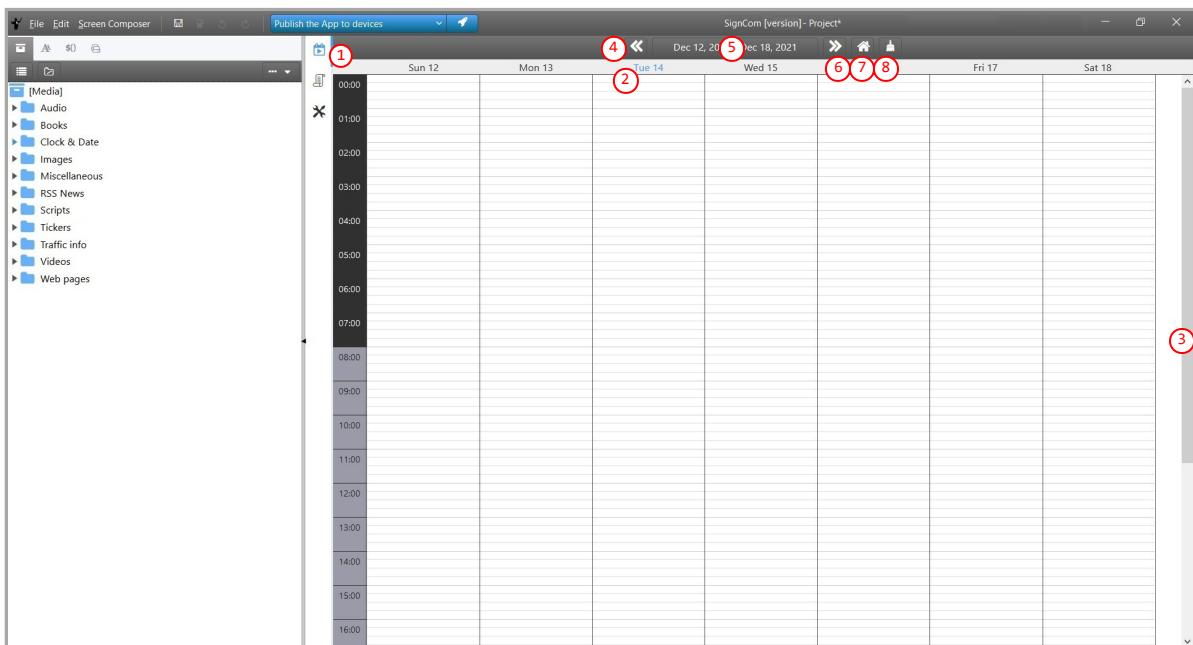
The current day (2) is highlighted in blue style.

With the scrollbar, scroll to the top or to the bottom (3) to view the calendar time slot from 00:00 to 24:00.

The calendar toolbar support these buttons:

-  (4) allows to view the `channel programmation` of the previous week,
-  (5) allows to view the `channel programmation` of the next week,
-  (6) allows to view the `channel programmation` of the current week,
-  (7) allows to delete all past `calendar time slots` except those having recurrence with the current day or those having recurrence with a date in the future.

The week start date/end date (5) input allows to view the `channel programmation` at a wished week in the past or in the future.



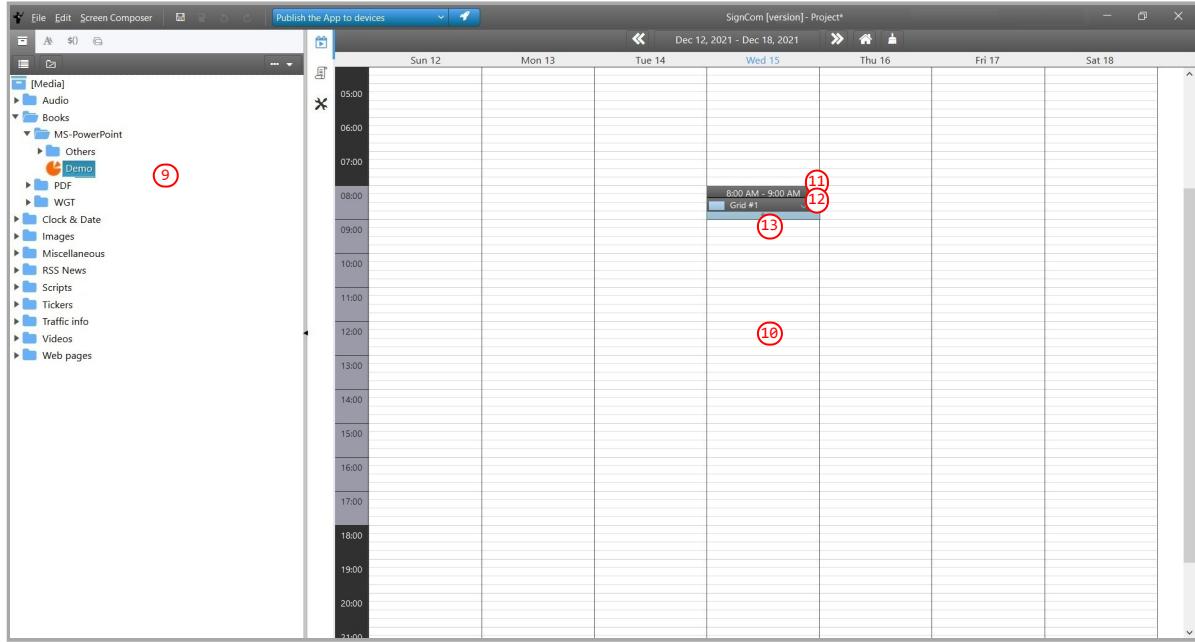
Calendar time slot creation

After having selected the appropriate calendar week, drag'n drop a media from the **Media** tab (9) to the calendar (10) at the appropriate date and start time of a time slot:

- if the calendar time slot (11) is not already existing, it is automatically created with the dropped media inside,
- the display grid inserted by default in a new calendar time slot is the default display grid (12).

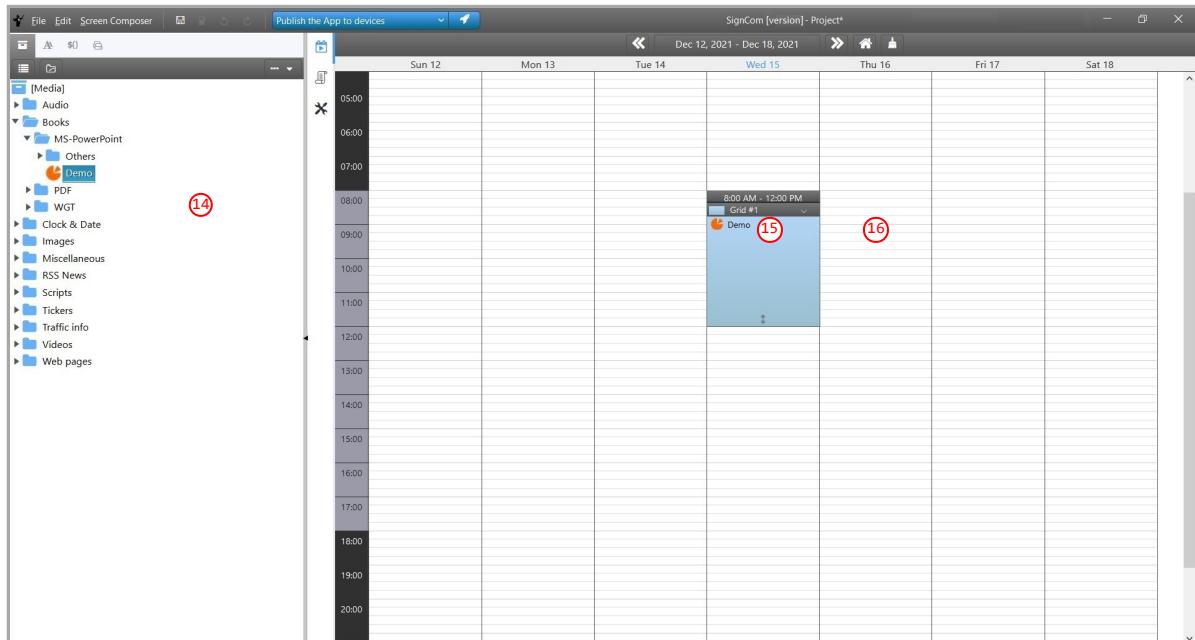
Using the mouse, extend the calendar time slot (13) by dragging the bottom of the calendar time range downwards with the mouse pointer.

It is also possible to start it before by dragging the top of the calendar time range upwards with the mouse pointer.



Drop again some medias from the **Media** tab (14):

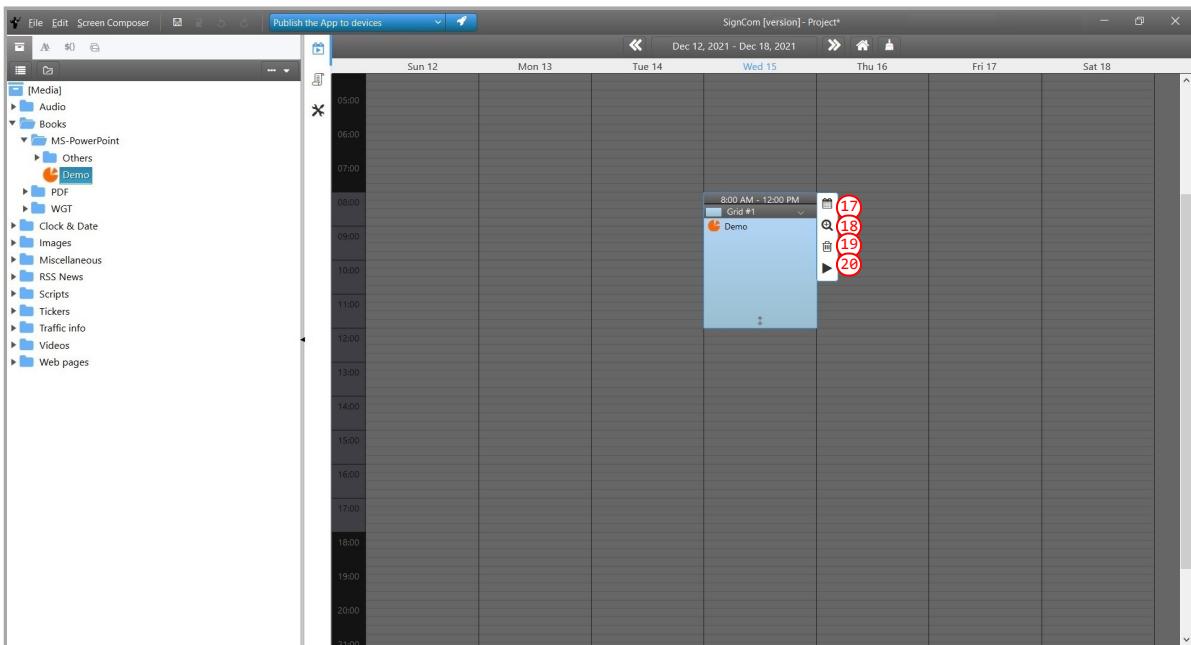
- to the calendar time slot (15) to abond the media sequence ,
- to other date and start time (16) of the calendar to create new calendar time slots .



When dropped in a new calendar timeslot, the media is inserted in the *main display region* in a scene driven by the default *display grid*.

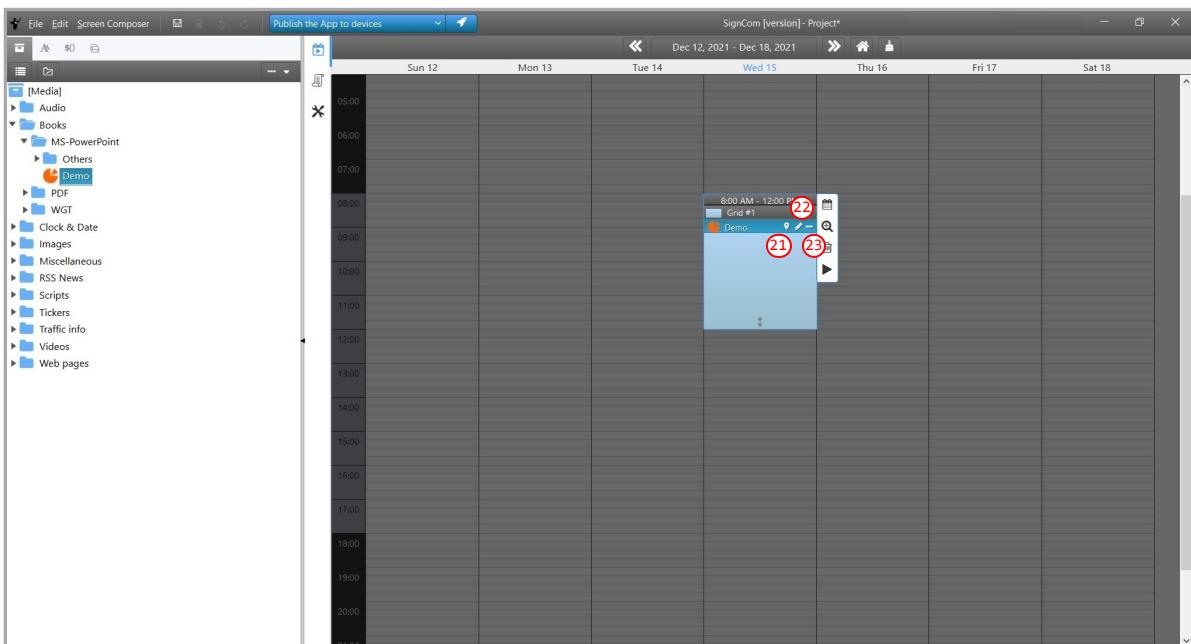
Select the calendar time slot to make appear its contextual menu supporting the following buttons:

- Edit the time slot planning **(17)** allowing to open the planning window,
- Edit the media sequence of the time slot **(18)** allowing to open the window to view the media sequence in the different display regions of the calendar time slot ,
- Delete the time slot **(19)** allowing the delete the entire calendar time slot ,
- Preview the time slot **(20)** allowing to launch the calendar time slot preview window.



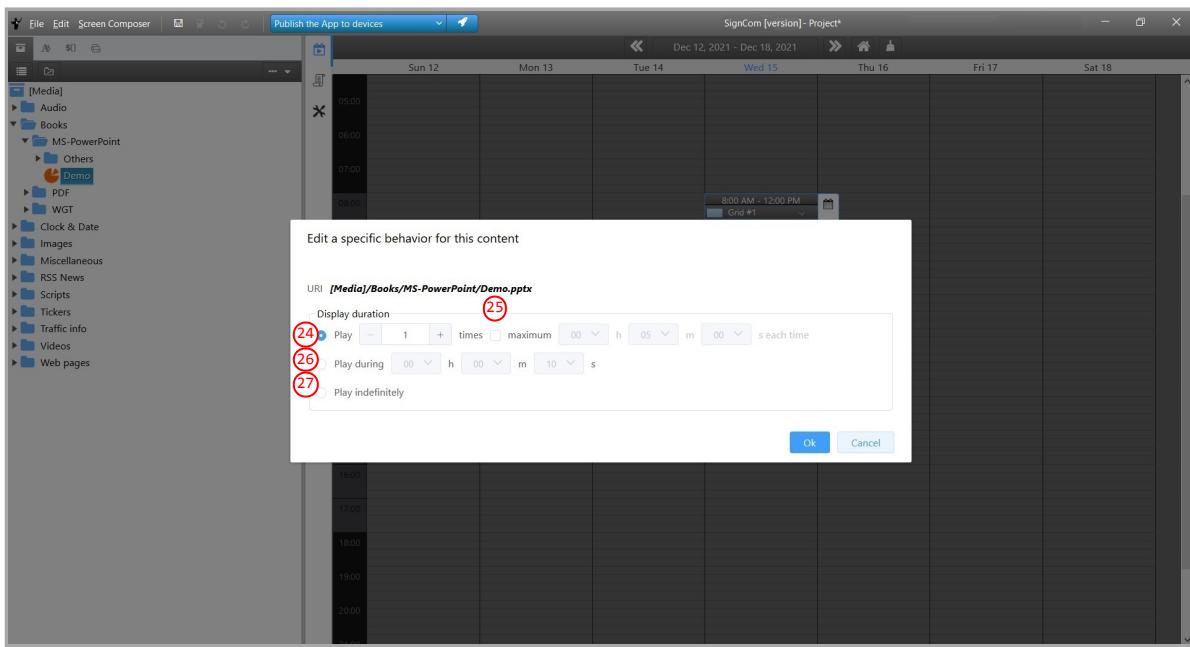
Fly the mouse over the Demo (.pptx) media to make appear the following buttons:

- Reveal **(21)** allowing to expand the media file tree in the Media tab ,
- Edit the specific behaviour for this content **(22)** allowing to edit the playback behaviour for this media,
- Remove **(23)** allowing the remove the media from the media sequence .



Click on the **Edit a specific behaviour for this content** button to edit the specific behaviour whose value can be:

- **(24)** Play the media once or several times,
 - **(25)** as option, set a maximum duration HH:MM:SS for each the times that the media is played,
- **(26)** Play during HH:MM:SS the selected media,
- **(27)** Play indefinitely the media allowing to play the selected media several time with never ending.

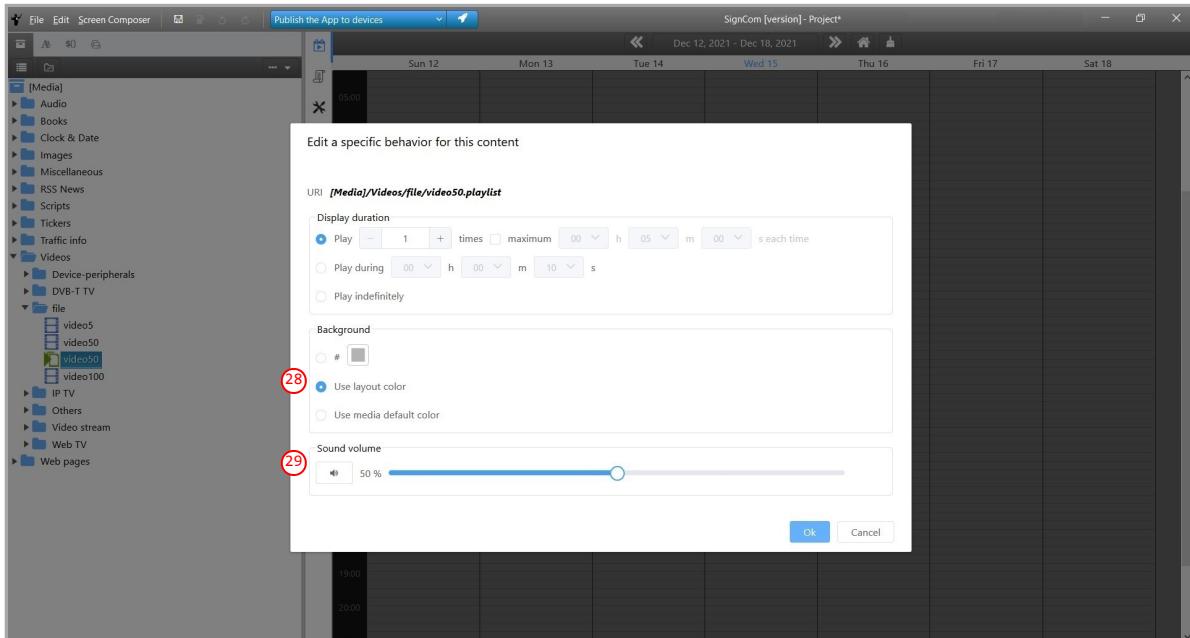


Add a `video50.playlist` media to your existing calendar time slot press the **Edit a specific behaviour for this content** button again for this media.

Depending on the type of selected media, the specific playback behaviour can support also some button, for example:

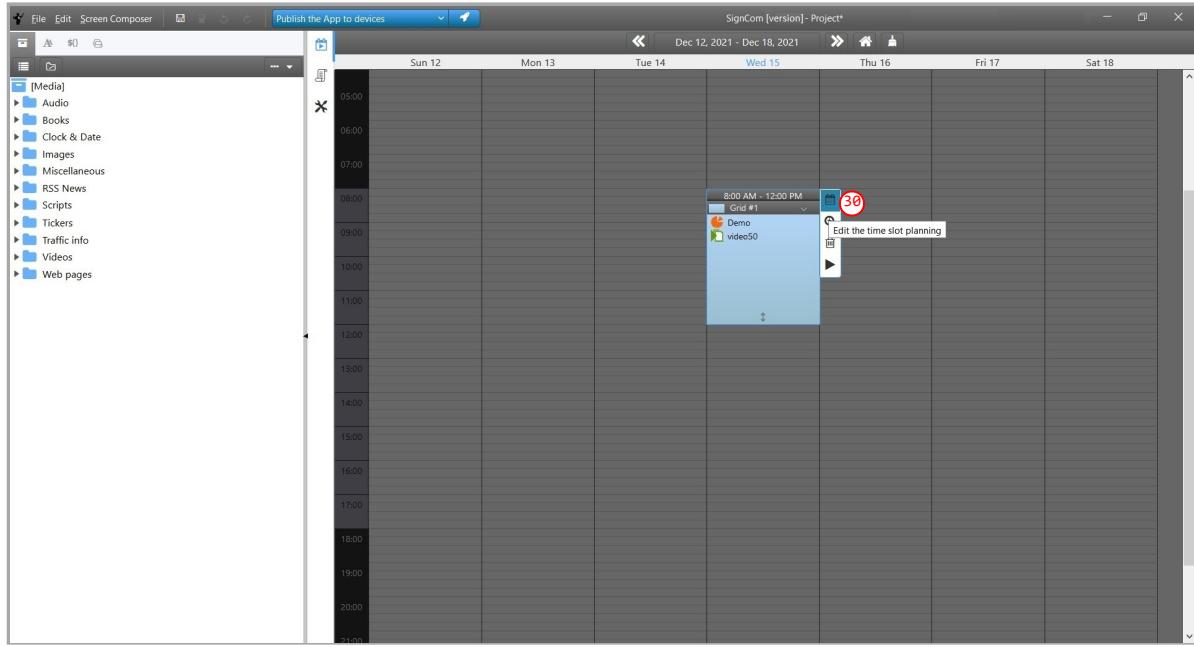
- for playlists and playfolders:
 - set the background color **(28)**,
 - set the volume **(29)**,
- for URIs:
 - set the HTML simulated width allowing to zoom in on the Web page content.

Example of playback behaviour window for a `.playlist` media (e.g. `Video50.playlist`):



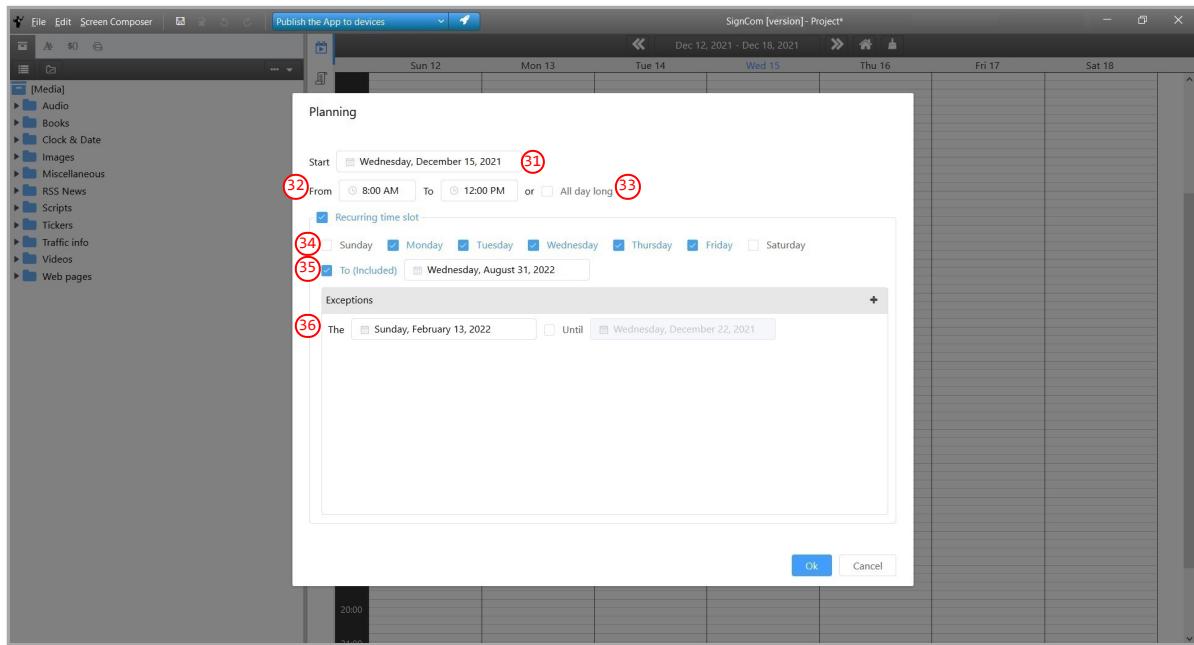
Edit the calendar time slot planning

Click on **Edit the time slot planning**  button to configure the planning attributes of the calendar time slot.

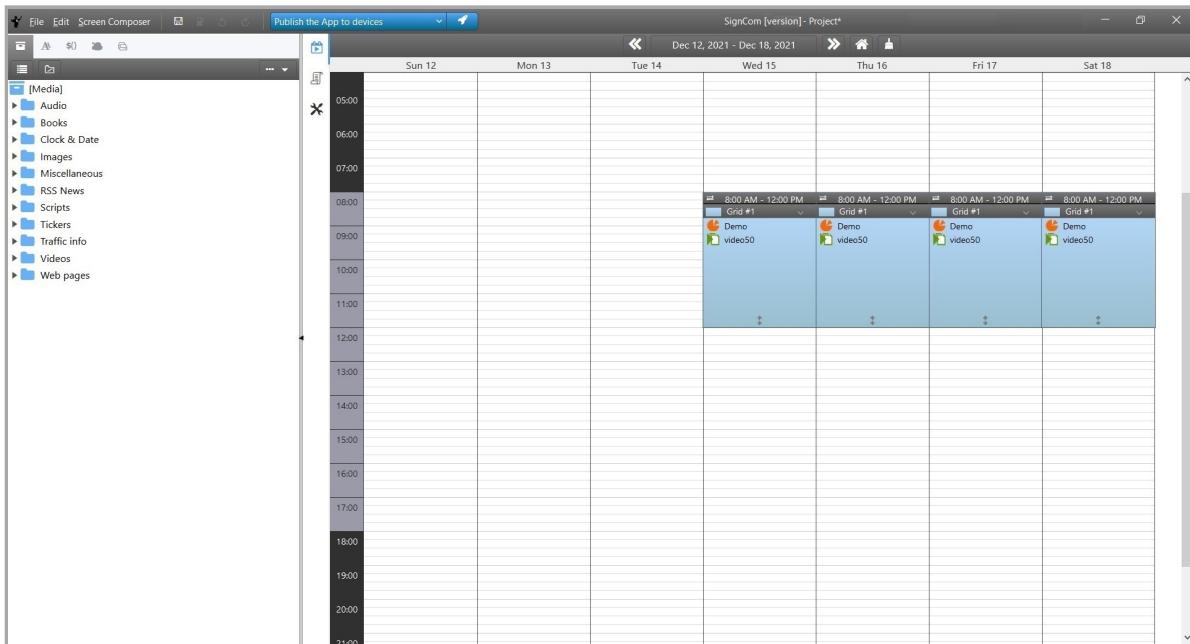


It allows to define:

- **③1** start date : start date of the calendar time slot ,
- **③2** the start time and end time of the calendar time slot ,
 - **③3** selecting *All day long* configure the calendar time slot from 00:00 to 24:00,
- **③4** the recurring days allowing to support recurring calendar time slot ,
 - **③5** the recurring end date allowing to set an end date to the recurring calendar time slot ,
 - **③6** the exceptions allowing to define the specific days when the recurring calendar time slot is not programmed.



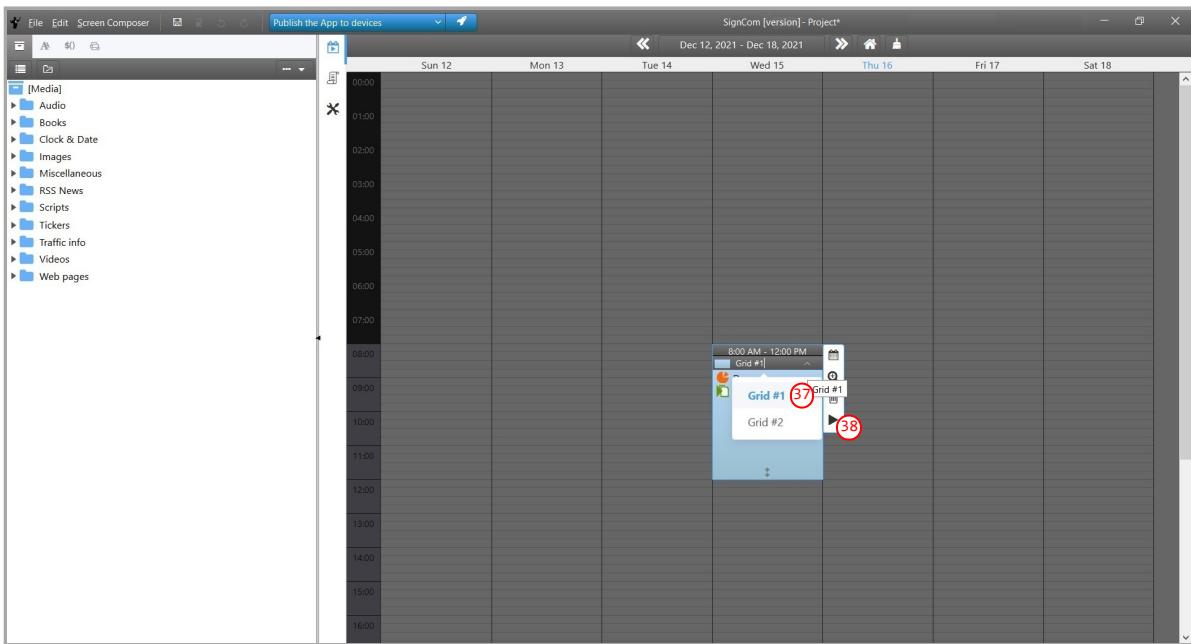
Example with a recurring event:



Preview of the calendar time slot

If your calendar time slot has multiple scenes, select the appropriate scene (37). If the user does not select a particular scene, the preview of the calendar time slot will start on the default scene.

Then click on the ► Preview the time slot (38) button to launch the preview window for this calendar time slot.



The different scenes can be created and modified in the next chapter § [Edit the media sequence of the calendar time slot](#).

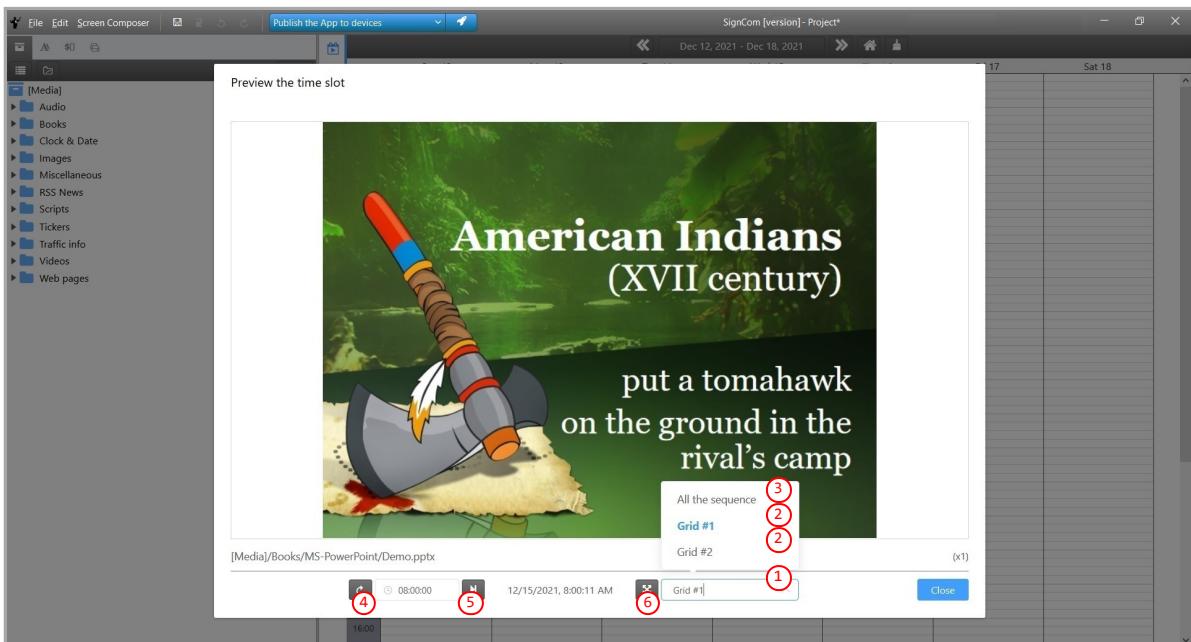
The preview window is launched for the default scene by default. Anyway, it is possible in this window with the Select the scene to preview ↗ (1) to:

- select a wished scene, driven by a display grid (2) to preview it,
- select the all the sequence (3) value to preview all the scenes, the one after the other.

The preview window supports also:

- a ⏪ restart (4) button,
- a ⏴ next (5) button to browse the medias of the selected scene or browse the medias of the all the scenes,
- a ⏷ expand (6) button to expand the preview window in full screen,
- a time input with the HH:MM:SS format to choose the time for the preview start time.

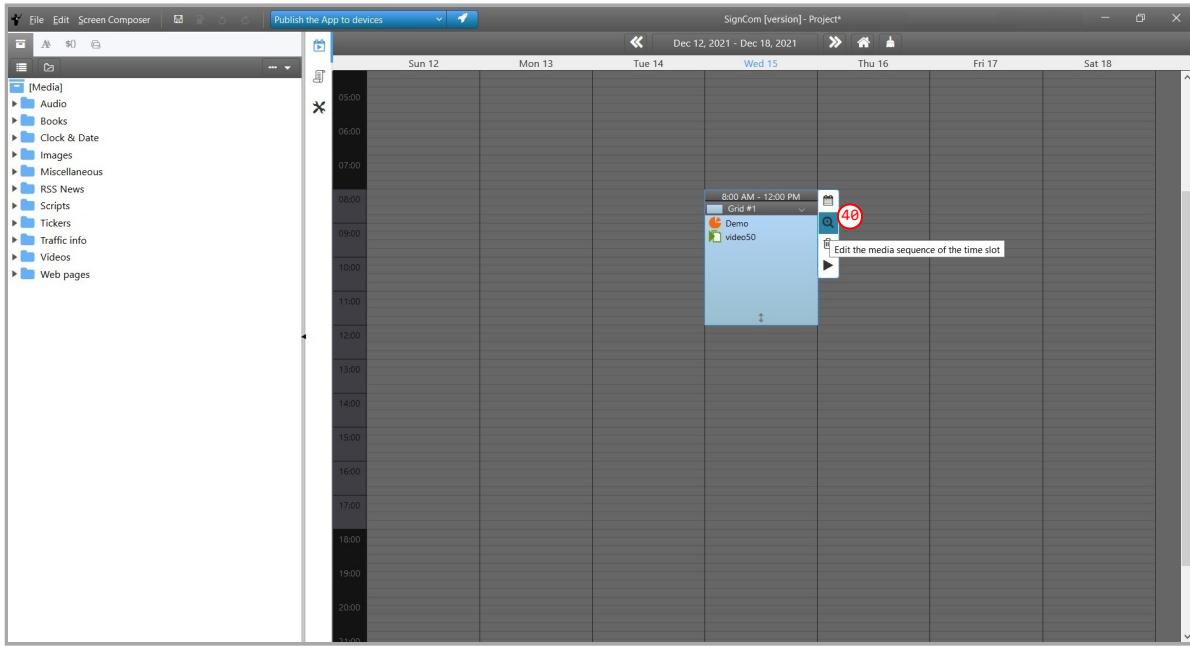
⚠ The time value of the Calendar time slot displayed by default when the preview window is launched is the start time of the Calendar time slot. Remember to enter the correct time here. The preview date is the date of the selected Calendar time slot.



Choosing the time for the preview allows to preview a scene at a defined time, useful for example to preview a scene having medias with validity date or media displaying content depending on the date (like the Agenda, MeetingRoom OR Calendar countdown HTML widgets).

Edit the media sequence of the calendar time slot

Click on the **Edit the media sequence of the time slot**  button to have an overview of the media sequence of the calendar time slot and abound medias in the different display regions .

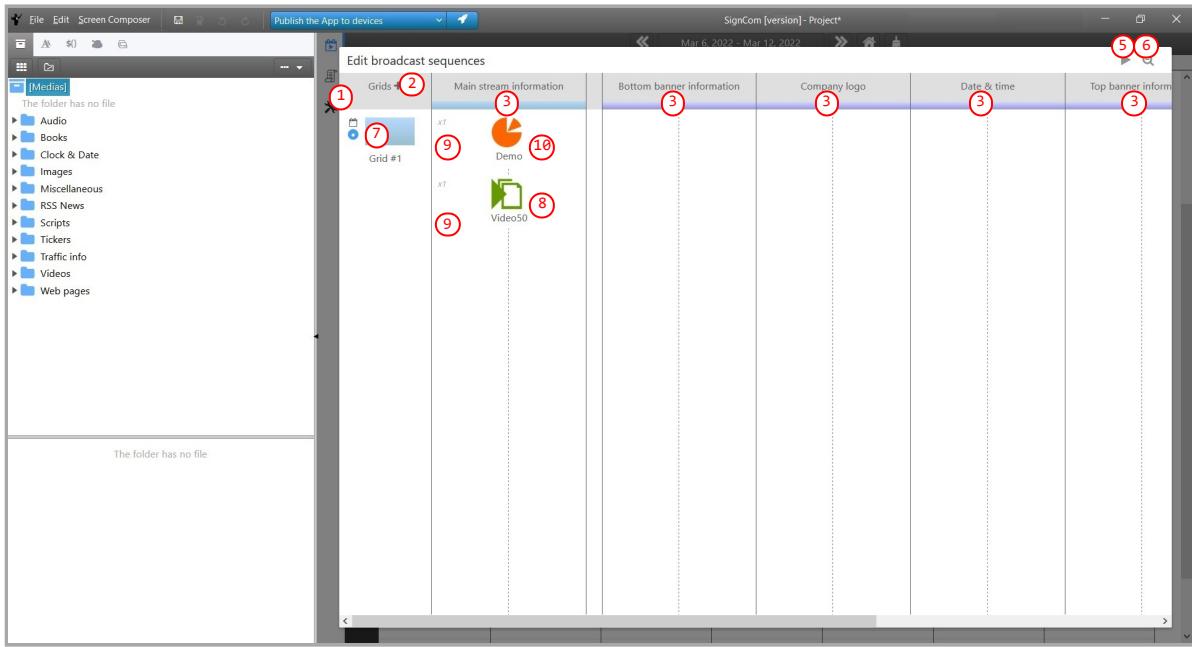


The media sequence window shows:

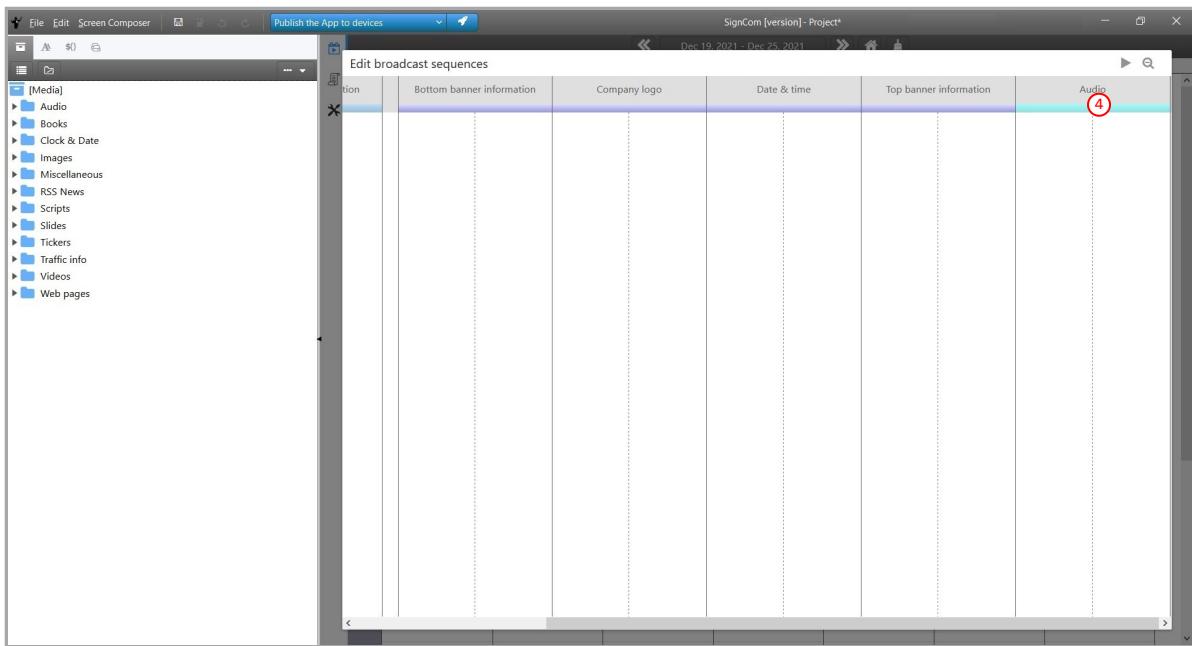
- a grid + column  displaying all the different scenes , driven each by a display grid , created for this calendar time slot ,
 - one column per display region   with their own media sequence ,
 - a   button allowing to add new scene , driven each by a display grid ,
 - a   preview the time slot  button allowing to preview all scenes in the calendar time slot , either one after the other with scene transitions, or only one scene ,
 - a   close button to close the window ,
 - the specific playback behaviour  per media:
 - x_1 ,
 - $hh:mm:ss$,
 - $x_1 \text{ max } hh:mm:ss$,
 - ∞ (for infinite) .
-  A column for audio region  with its own media sequence is also available when it has been created.
-  All the medias are played from the top to the bottom. When the playback at the bottom of the media sequence is finished, the SignCom App restart to play the first media at the top of the media sequence .

In the example:

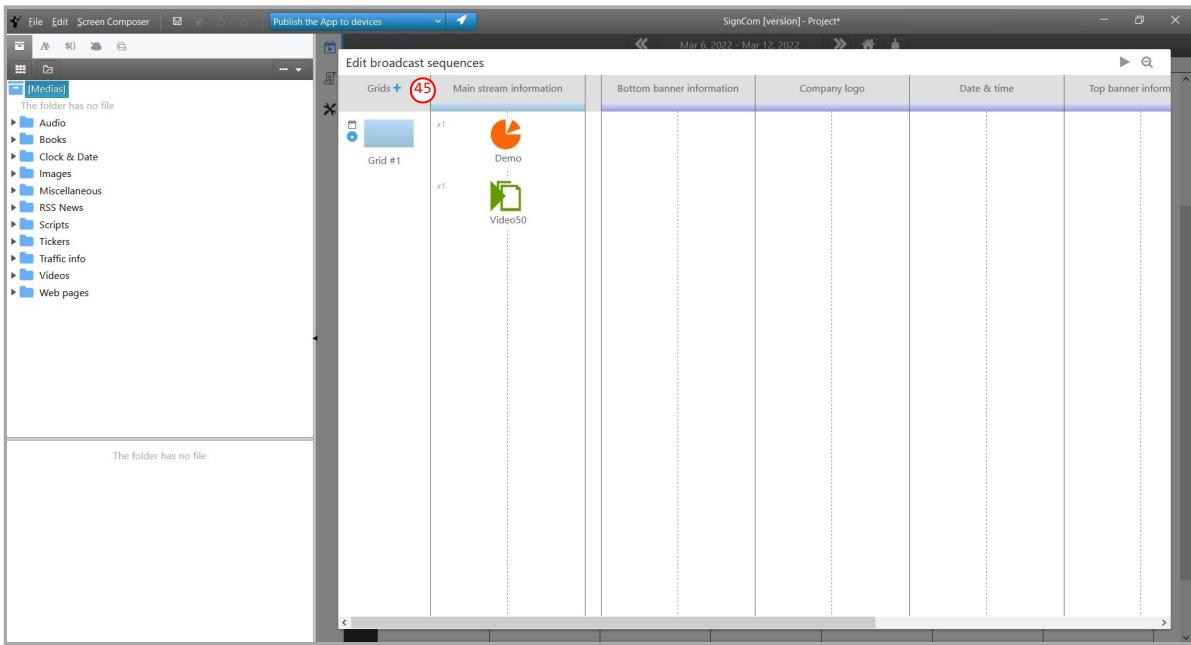
- only the `Demo` (.pptx) (10) and the `video50` (.playlist) (8) medias located in the `main` (2) display region have to be played in the `scene` (7) driven by the `Grid #1` display grid .
- the `media sequence` for other `display grids` are empty for the moment.



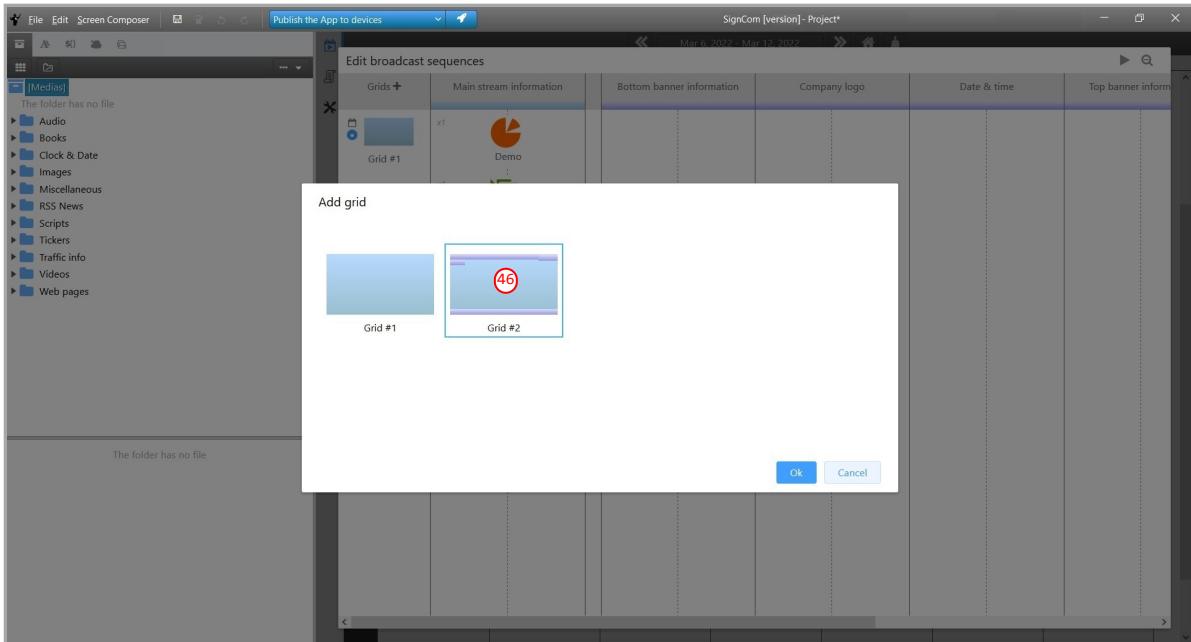
Scroll to the right to watch the `media sequence` programmation for other `display regions` , especially for the `audio` region (4).



To create a display grid transition, click on the Add grid  button.



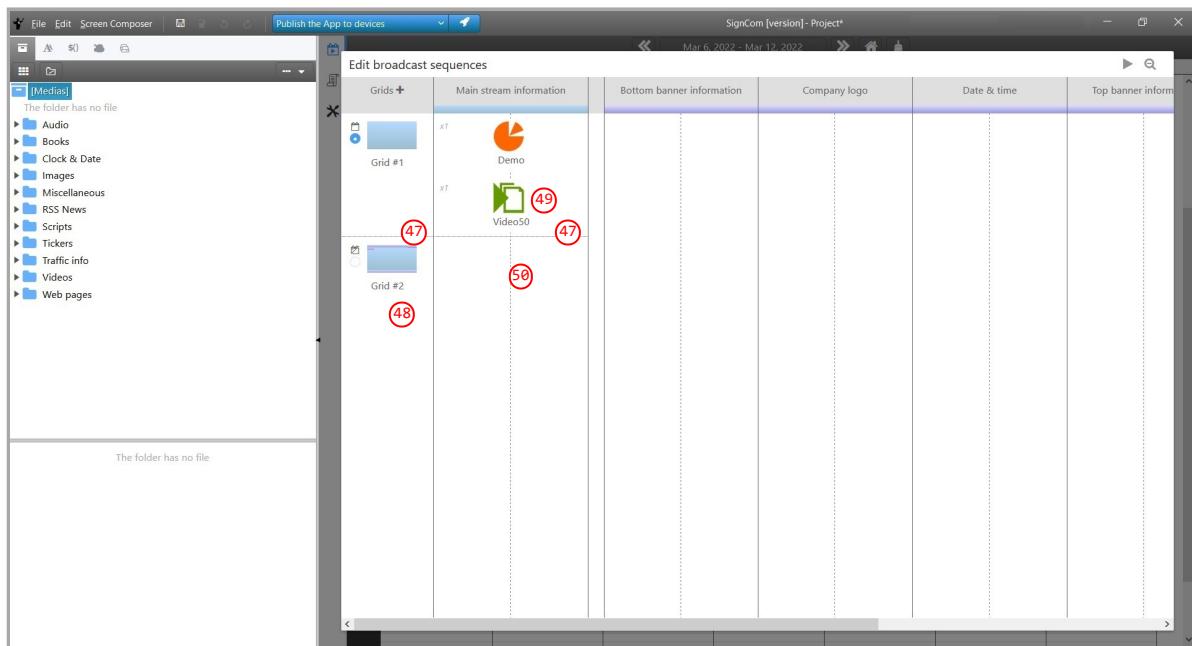
Select for example a display grid which is not yet used (e.g. Grid #2 ) and click on the ok button.



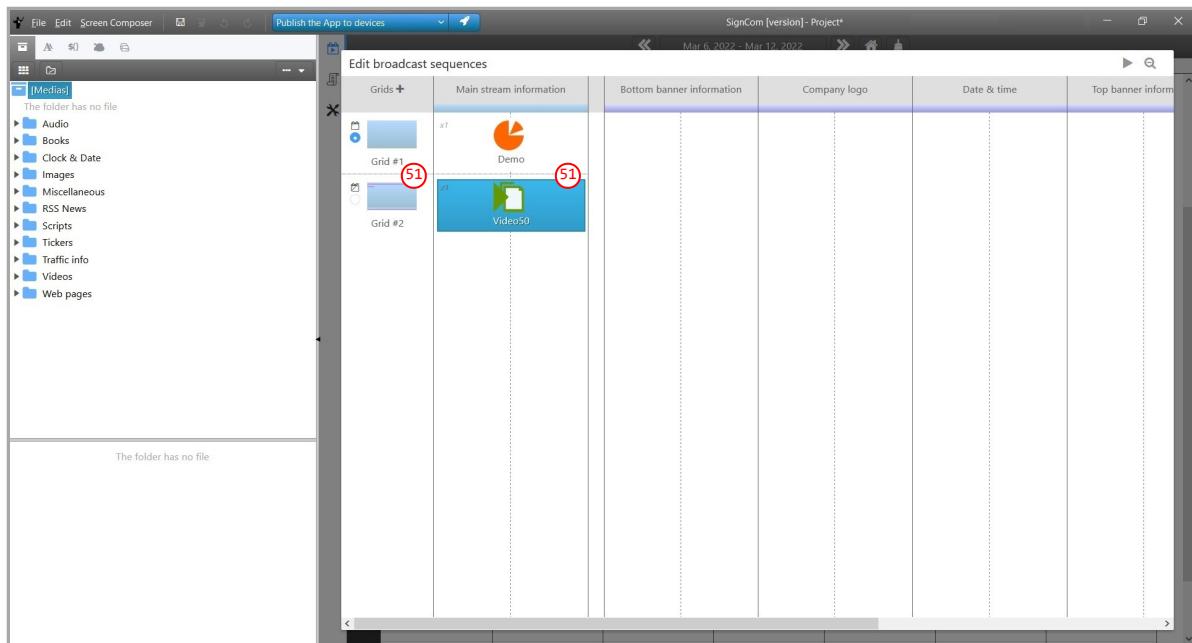
A scene horizontal splitter in a grey color has appeared, running from the Grid + column **(47)** to the *main* display region **(47)** (e.g. *Main stream information*).

So that the *scene* transition is consistent, the new *scene* **(48)**, driven by the *Grid #2*, must not stay empty.

Drop, for example, the existing *video50* **(49)** media from the *scene* driven by the *#Grid #1* display grid to the *scene* **(50)** driven by the *#Grid #2* display grid.



The scene horizontal splitter has moved towards the top **(51)**.

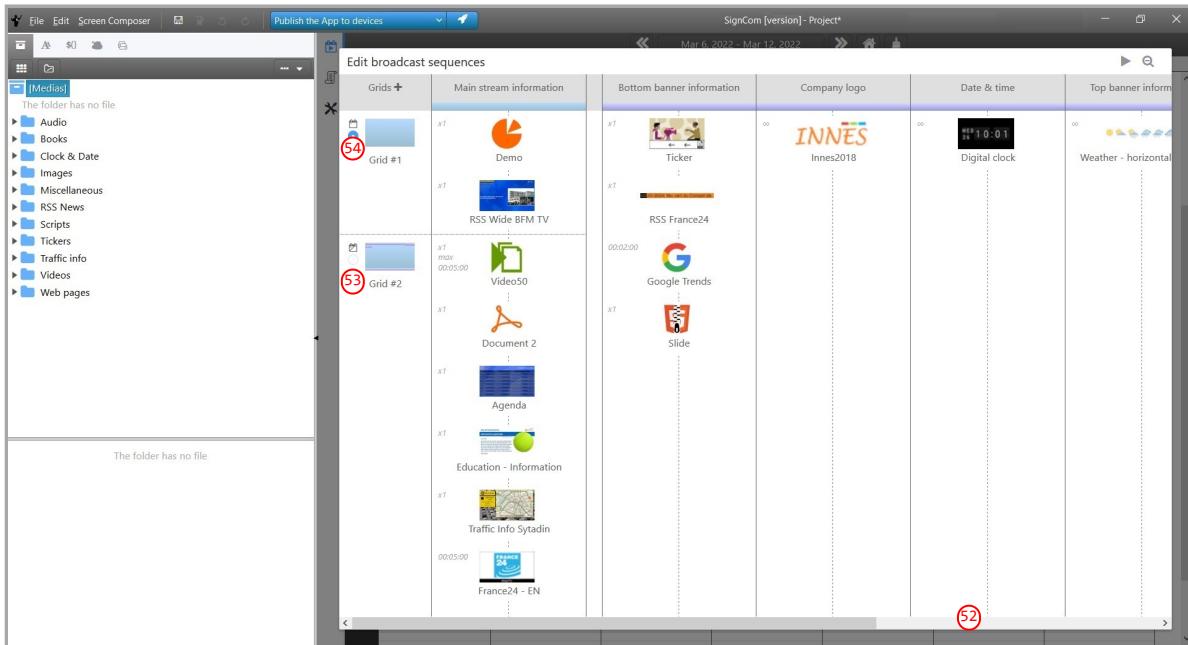


Drop then the appropriate medias in the appropriate display regions.

The display regions are usually abounded with medias having an appropriate aspect ratio. Insert rather:

- medias with 16/9 aspect ratio in display regions with 16/9 aspect ratio,
- medias with 9/16 aspect ratio in display regions with 9/16 aspect ratio,
- medias with horizontal banner aspect ratio in display region with horizontal banner aspect ratio,
- medias with vertical banner aspect ratio in display region with vertical banner aspect ratio,
- audio medias (audio files, audio playlist) in the audio regions, ...

Scroll to the right 52 to display the display regions that are not visible in the window.

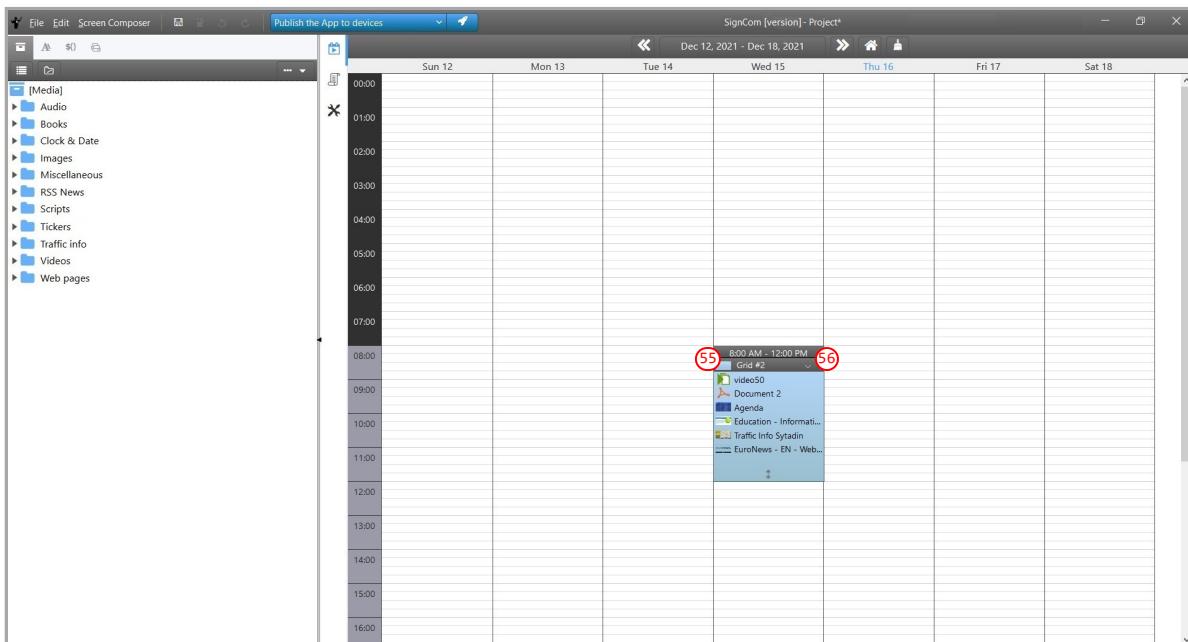


Check the media behavior value for each media in each zone:

- x1 ,
- hh:mm:ss ,
- x1 max hh:mm:ss ,
- ∞ (for infinite).

The display the grid in the calendar radio button allows to define the scene, driven by its associated display grid, selected by default in the calendar time slot:

- this pictogram 53 is displayed when this scene (e.g. Grid #2 * display grid label) is defined as default selected scene 55 in the calendar time slot ,
- this pictogram 54 is displayed when this scene (e.g. Grid #1 * display grid label) is not defined as default selected scene 56 in the calendar time slot .



The scene selected for the preview can be changed with the ▼ down arrow button.

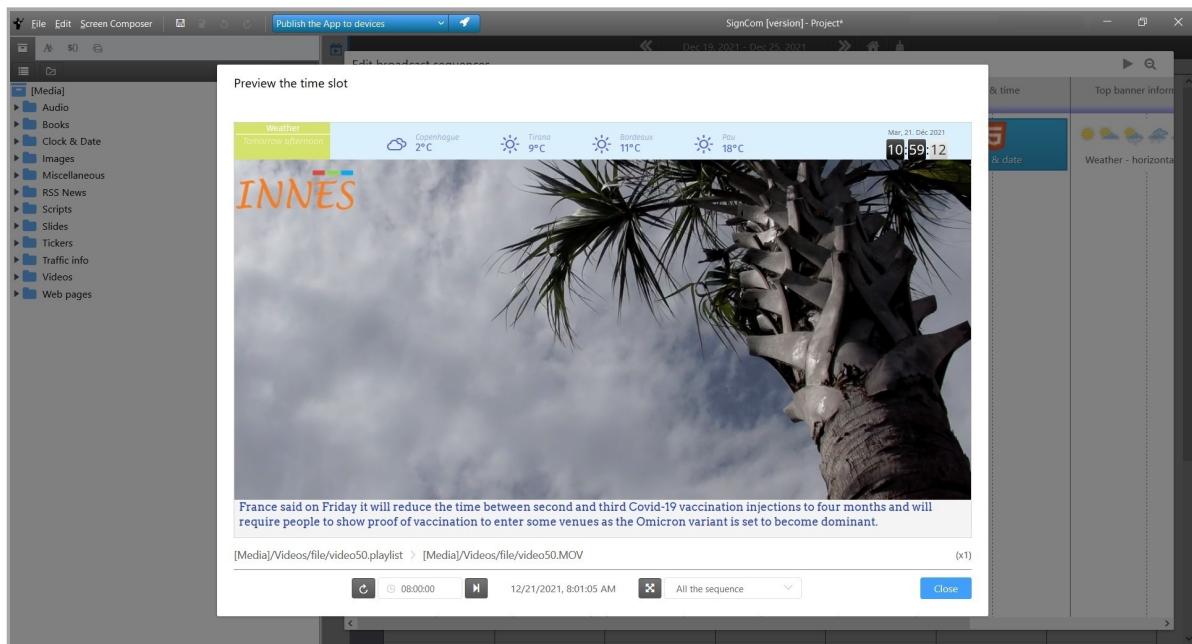
Click on the ► preview the time slot to launch the preview of all the sequence (all the scenes).

When launched from the media sequence window, only the preview of all the sequence (all the scenes) is available.

To preview only one scene, close the window, select a calendar time slot, select the appropriate scene (display grid label) and launch the preview from the contextual menu of the calendar time slot.



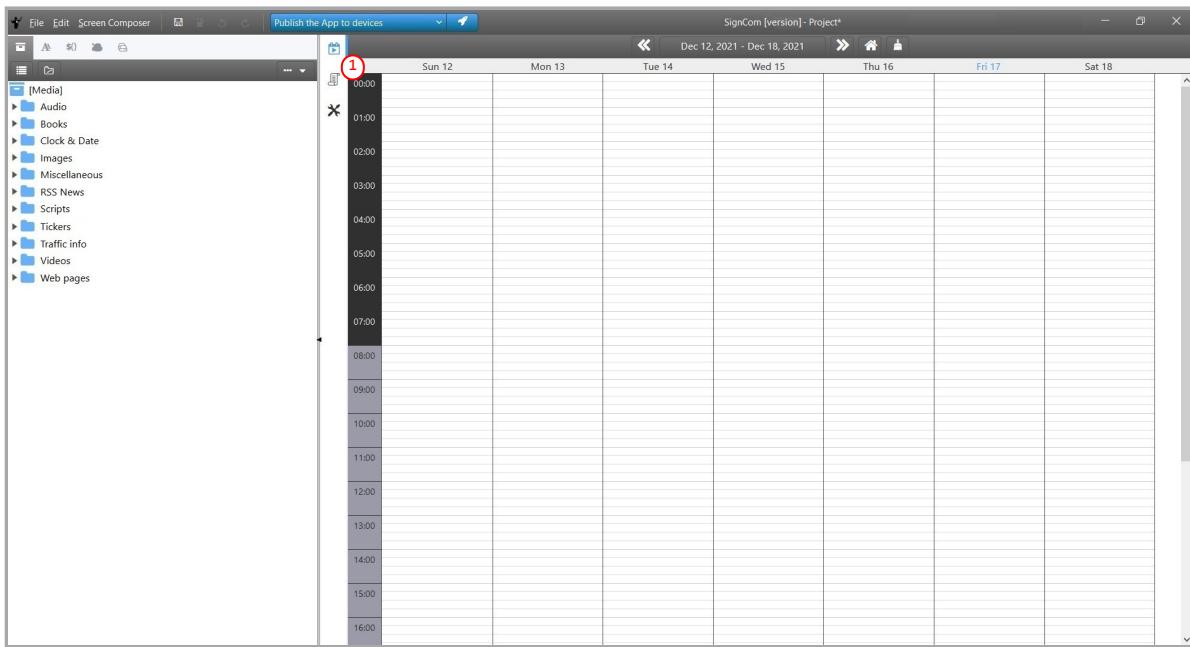
Click on the **Next** button to browse all the others medias of the `media sequence` in the preview window.



When your `channel programmation` is correct, publish the SignCom App on your devices.

3.8 SignCom > Manifest

Click on the `Manifest`  tab to open the `Manifest` pane.



List of medias and resources used, size and dependencies

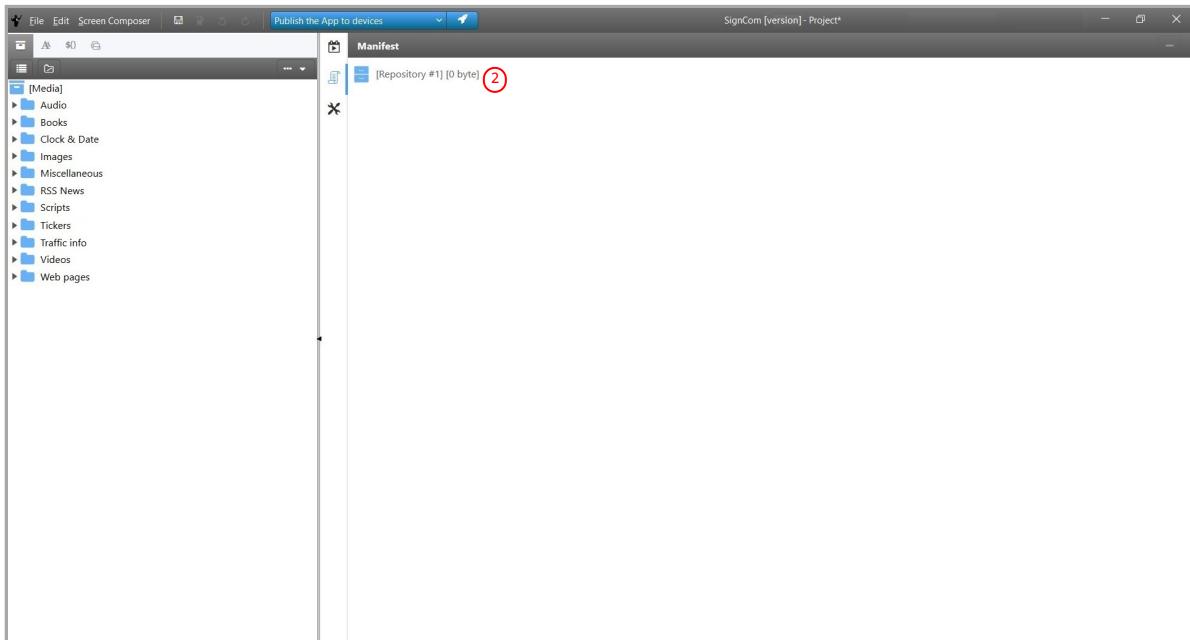
The `manifest` tab shows:

- all the medias and text files required by your App to work properly and,
- the overall size  (2) of the App content.

Indeed, to work properly, once published, the SignCom App needs all the required medias so that it can work 24/7/365. It includes all the medias contained in the calendar time slots :

- for all the previous weeks,
- for the current week,
- for all the next weeks.

When no `calendar time slot` is added in the `channel programmation`, the content of the manifest is empty and the manifest size is 0 byte.

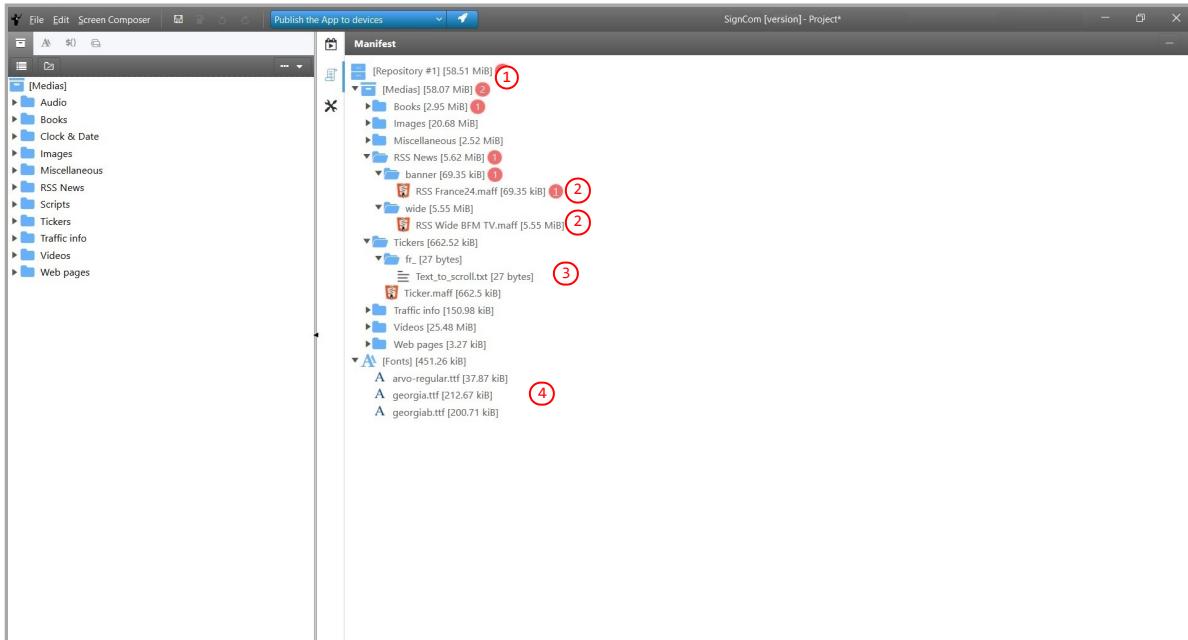


This is an example of manifest with a channel programmation containing calendar time slots with media. These files come from:

- medias inserted in the calendar of the channel programmation ,
- the configuration of a default media to a display region ,
- the support for a media as background image,
- the configuration of a media as default audio content ,
- the configuration of a media as default visual content .

The manifest tab shows also:

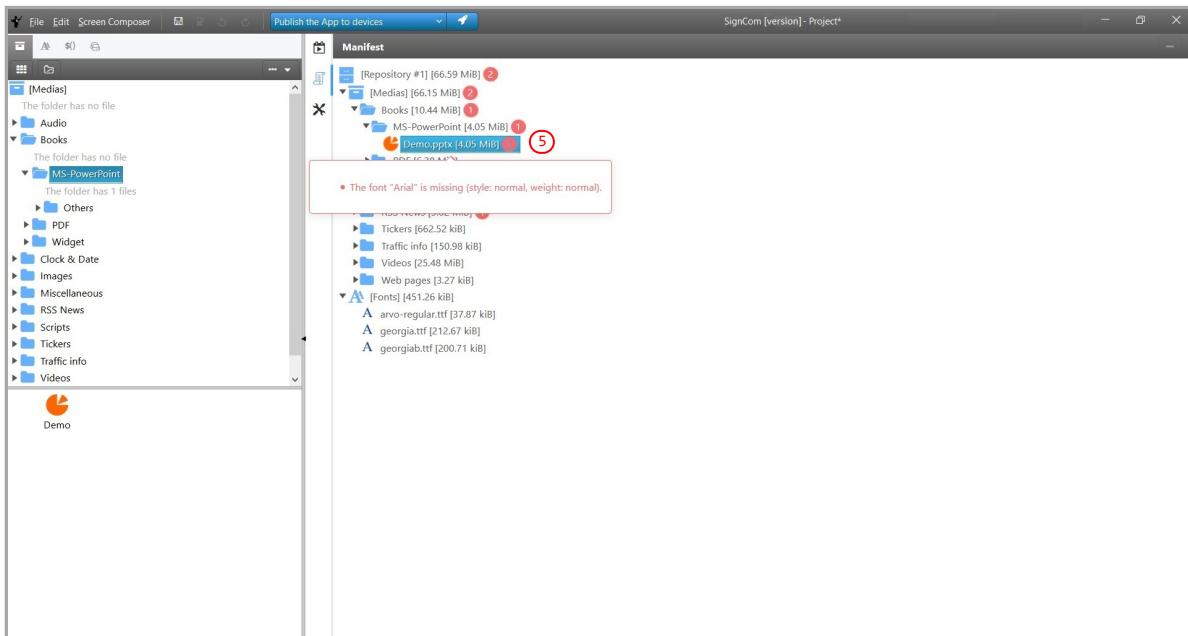
- the size (2) of each medias required by your SignCom App ,
- all the resources dependencies (text files (3), fonts (4), ...) required by the SignCom App to play the media properly.



When possible, it is advised to remove regularly from the channel programmation, the calendar time slots of the past week to avoid to embed medias of obsolete calendar time slots .

In case a dependency cannot be resolved, an additional red stamp (5) is displayed showing that the media may not work or the rendering may not be totally correct due to file missing or font missing. Click on the red stamp to display some additional information on the dependency that cannot be resolved.

⚠ The font stamp error is cleared from the manifest as soon as one font of the font family required by the media is found in the fonts library.



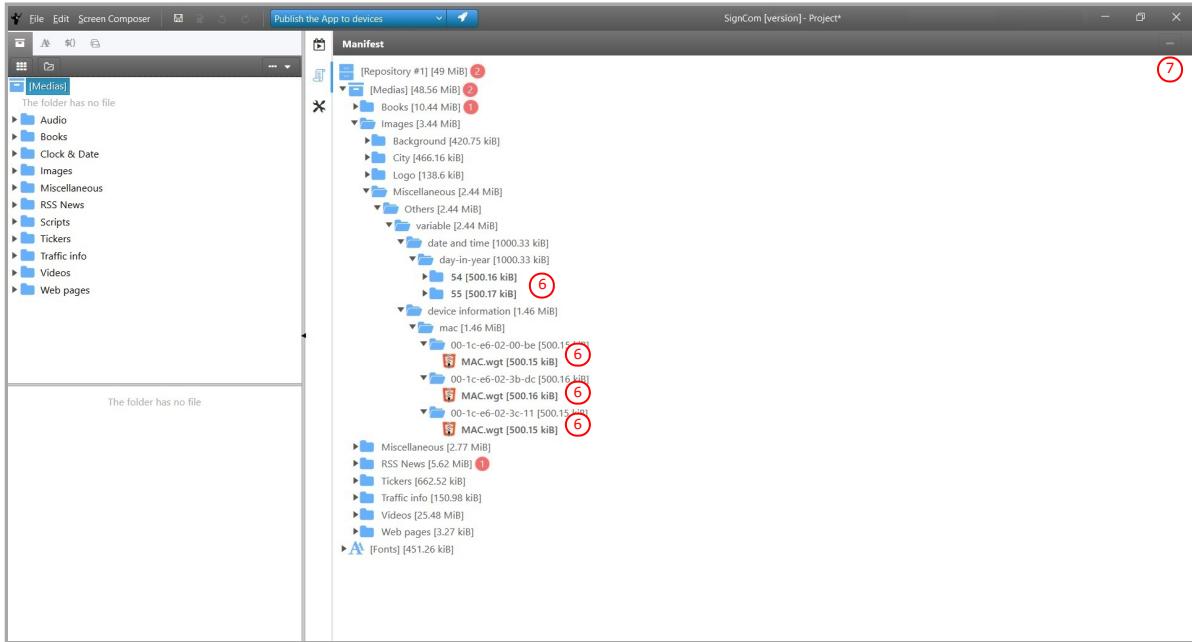
Additional files added by the user

The user can manually add some files to the SignCom App content by drop files from the `Medias` tab to the `Manifest` pane.

If your App has playfolders or URIs using `Device information` variables or `Date` and `time` variables, it is required to drop the suitable target folder or the suitable target files to the manifest else the device won't be able to find the content to play.

The files added manually in the `Manifest` pane appear in **bold** style (6).

To remove a file or folder in bold style added by the user, select it in the manifest and click on the Remove (7) button.



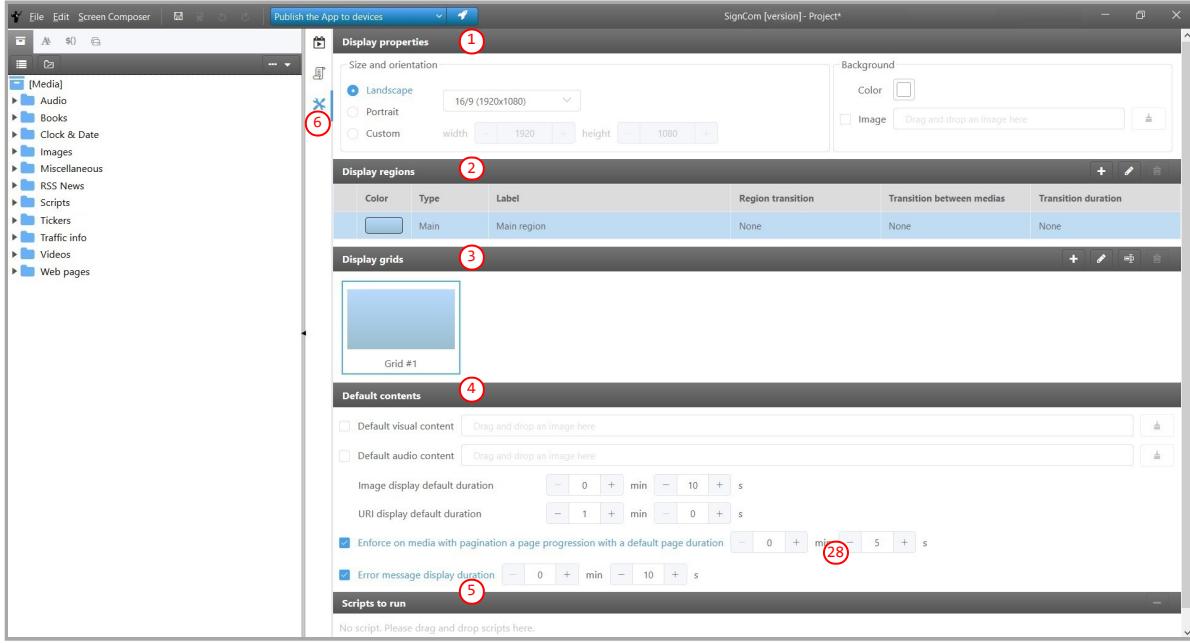
Remote file and remote folder download task

The App content update by remote file or remote folder download task, supporting the `http://`, `https://` and `smb://` schemes is not supported in Screen Composer G4.

3.9 SignCom > Properties

Click on the **Properties**  tab allows to configure:

- the display properties ,
- the display regions ,
- the display grids ,
- the default contents and default behaviours ,
- the scripts to run .



Display properties

The **Display properties** (1) part defines the `SignCom` display configuration. It often corresponds to the target screen configuration:

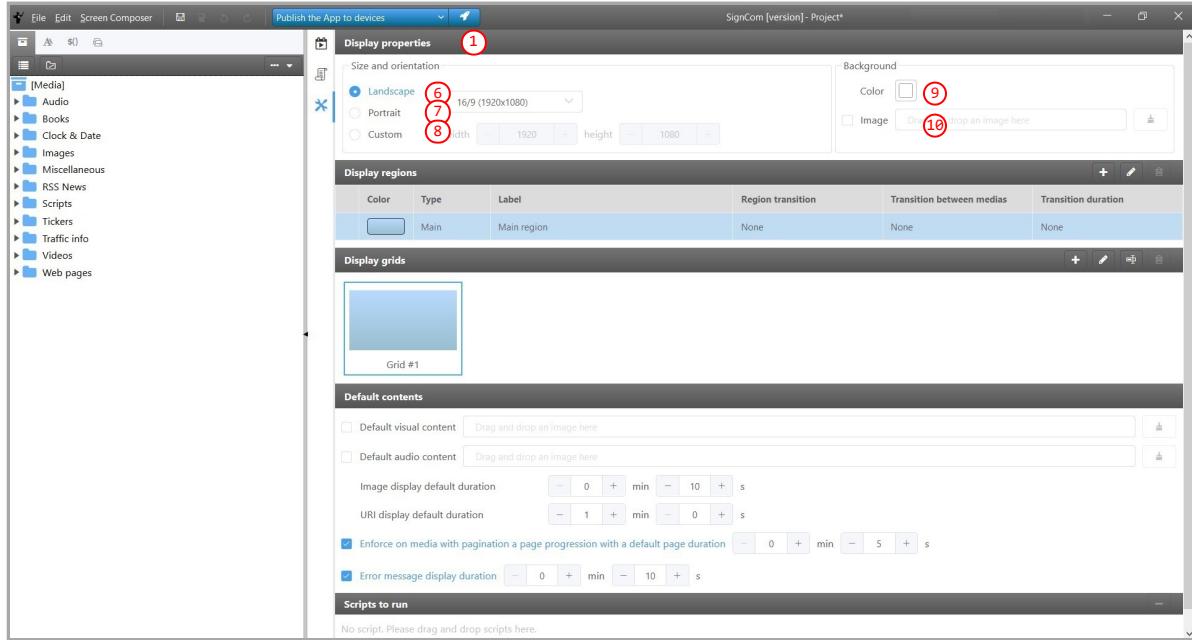
- **Size and orientation:**
 - **Landscape** (6),
 - **Portrait** (7),
 - **Custom** (8).

☞ Your project can handle either portrait or landscape screens. If you have both types of screens to handle, you need two separate project types, one for each screen orientation.

The **Background** part allows to define the `background` content that can be watched when the media does not fill the entire width or height of the screen or when an information message is displayed while the media cannot be played. It consists in:

- a background color (9), white color by default,
- an optional background image (10).

To set a background image, drag'n drop an image from the **Media** tab to the background image input (10).



☞ In case you configure the SignCom App in portrait mode, check also that the media players output are also configured in portrait mode (90° rotation).

Display regions

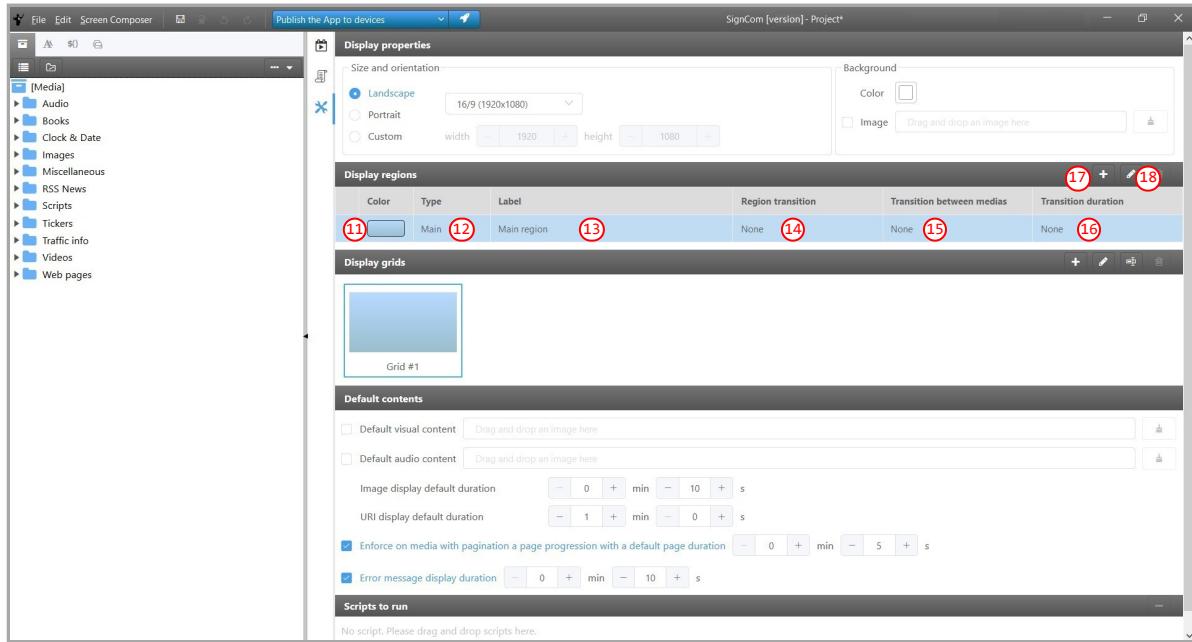
This part allows to display a summary of the available `display regions` for this project.

A `display region` allows to play a `media sequence` in a particular region of the screen.

By default, there is only one available `display region` called `main` having these characteristics:

- Color (11): *light Blue*,
- Label (12): *Main region*,
- Type (13): *Main*,
- Region transition (14): *none*,
- Transition between medias (15): *none*,
- Transition duration (16): *none*.

To edit a `display region` to change its attributes, click on the `Edit region` (18) button.

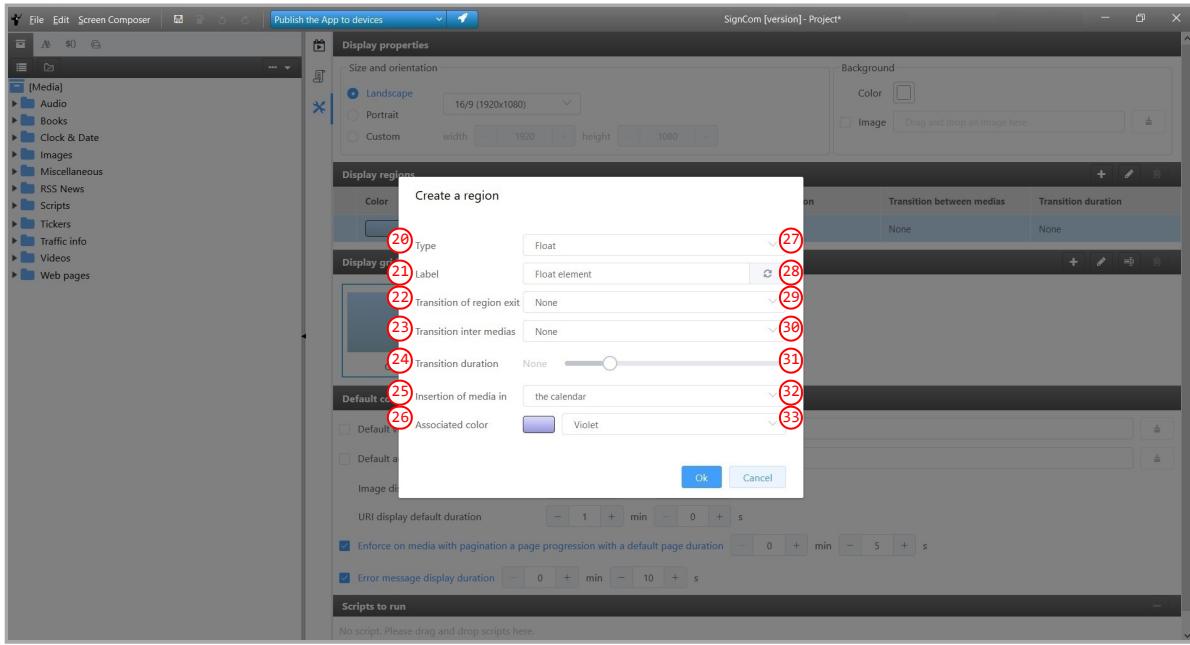


To display several others medias at the same time in a particular region of the screen, it is required to create others `display region` (e.g. banner for scrolling information, channel logo or company logo).

To create an additional display region, click on the  Create a region  button.

For any new display region, it is possible to define:

- the **Label** 
- the **Type** 
- the activation of **Region transition** 
- the activation of **Transition between medias** 
- the **Transition duration** 
- the **Insertion of media in** :
 - *the calendar,*
 - *the layout,*
- the **Color** 



Click on the refresh button to generate an automatic label depending on the chosen type of display region.

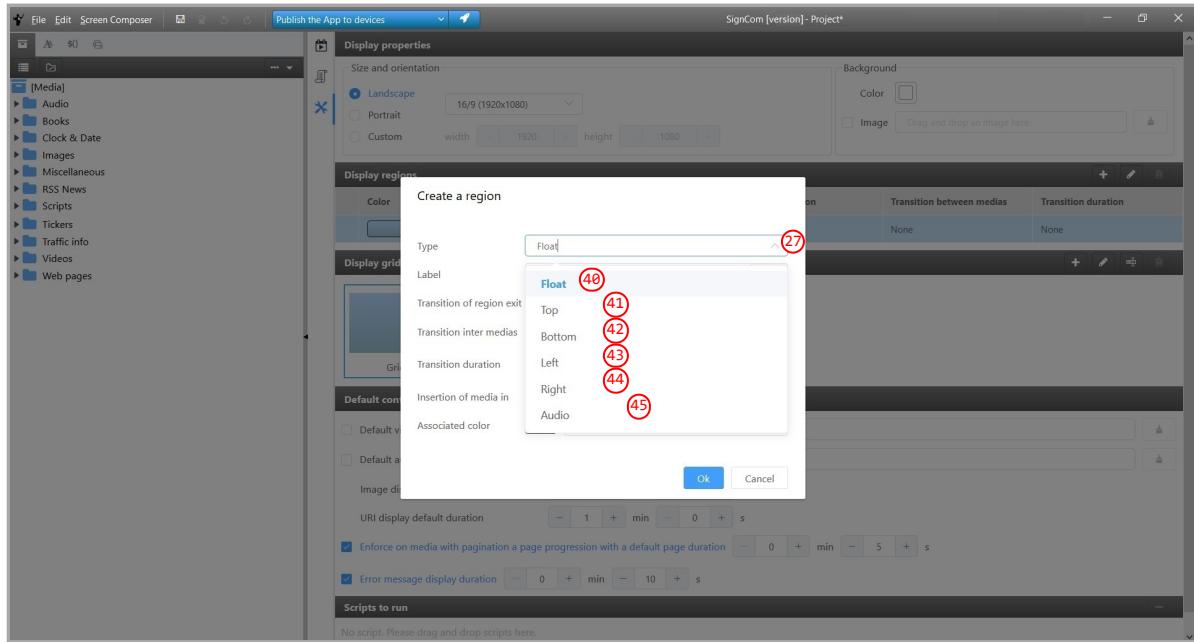
Select the down arrow button of the color to change the color value for this display region.

Select the down arrow button of the Type to change the type of the display region.

Three types of display regions can be added to the existing Main region:

- the standard visual regions with banner format which do not support overlap:
 - top ⑩: horizontal banner region stuck to the top of the screen, filling the screen width and having an adjustable height,
 - bottom ⑪: horizontal banner region stuck to the bottom of the screen, filling the screen width and having an adjustable height,
 - left ⑫: vertical banner region stuck to the left of the screen, filling the screen height and having an adjustable width,
 - right ⑬: vertical banner region stuck to the right of the screen, filling the screen height and having an adjustable width,
- the float ⑭ visual regions: regions with no spatial constraints (x & y coordinate, width, height), supporting region overlap thanks to their affected zindex (kind of calc depth),
- the audio ⑮ region: allows to play some audio content in parallel of the visual content.

Except for specific usage, it is recommended to use float display regions supporting overlap for banner regions.



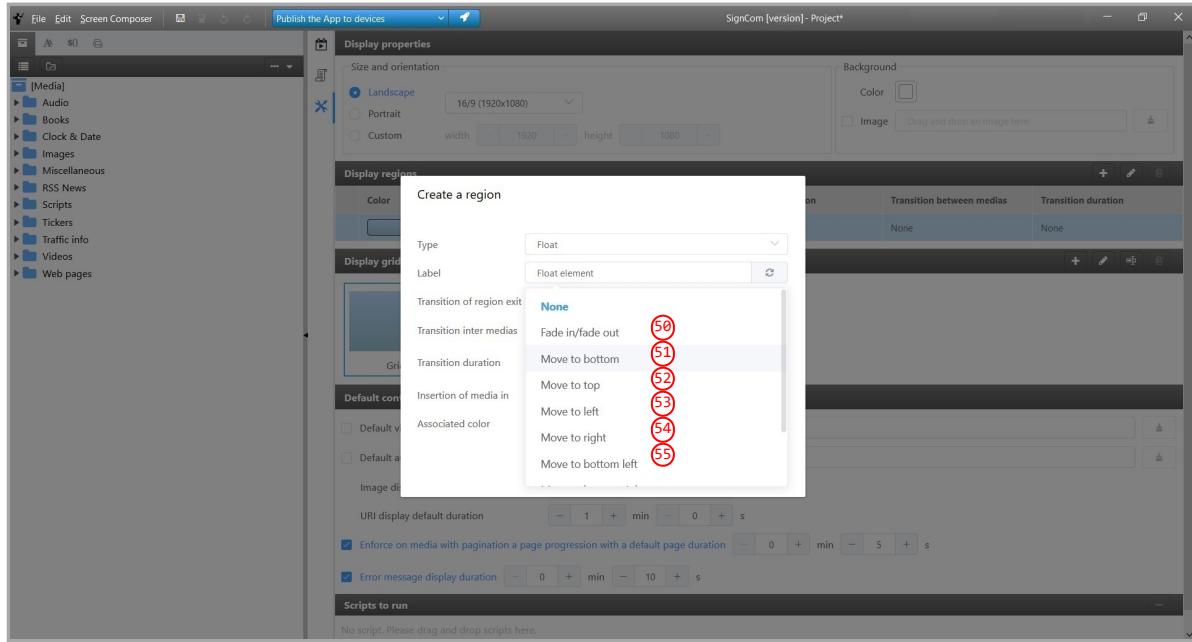
The region transition feature allows to activate progressive transitions between display region, typically while:

- display region disappearing,
- display region appearing,
- display region widening,
- display region narrowing.

Select the ▾ down arrow button of the region transition (29) to change the region transition value among the following:

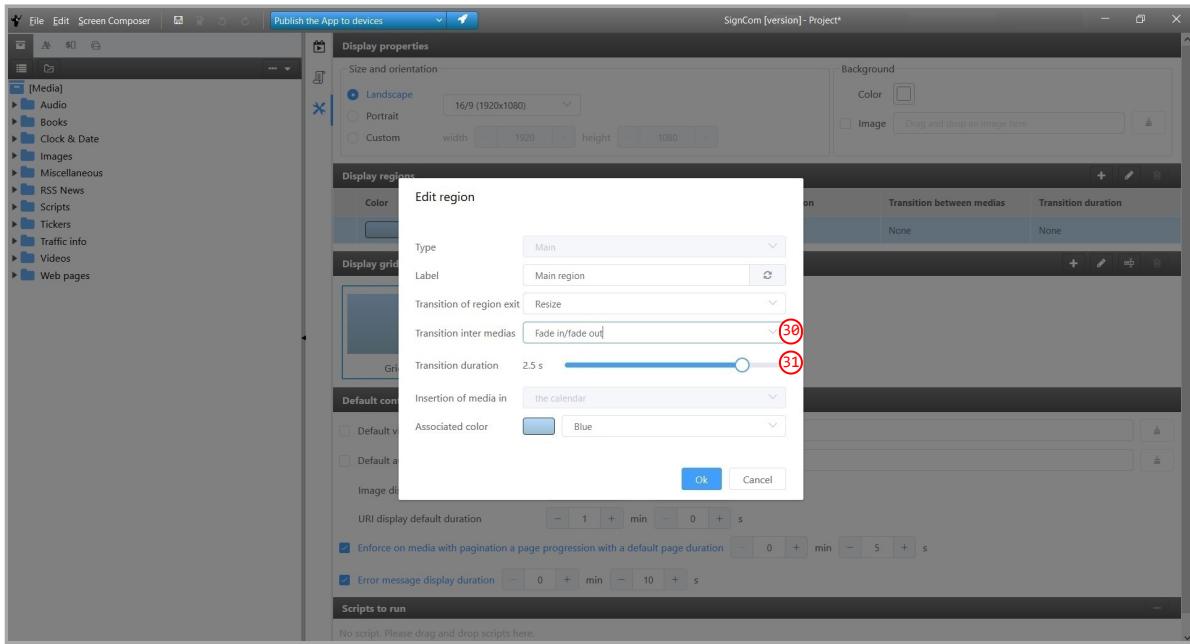
- *Fade in /fade out* (50),
- *Move to the bottom* (51),
- *Move to the top* (52),
- *Move to the left* (53),
- *Move to the right* (54),
- *Move to the bottom left* (55),
- *Move to the bottom right*,
- *Move to the top left*,
- *Move to the top right*,
- *Resize*.

 Depending on the type of display region, some region transition values may be not available.



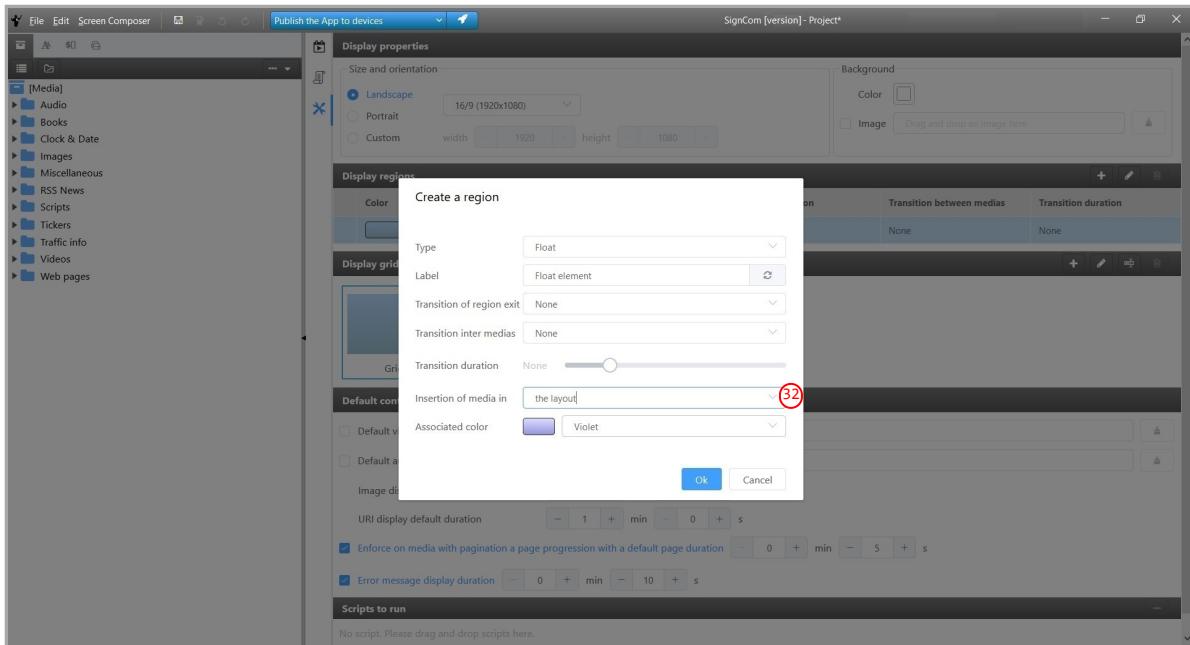
The transition between media feature allows to activate fade out/fade out transitions between the end of a media playback and the start of a next media playback. Select ▼ down arrow button of the Transition between medias **(30)** to change the Transition between media value among the following: - fading in/fade out, - none.

When a transition value is selected, it is possible to set the transition duration **(31)** between 0 and 3 seconds.



Except for the main display region, it is possible to tell that a display region will have to play always a same and only media. In this case, for the Insertion of media in drop down list, select the layout **(32)** instead of the calendar value.

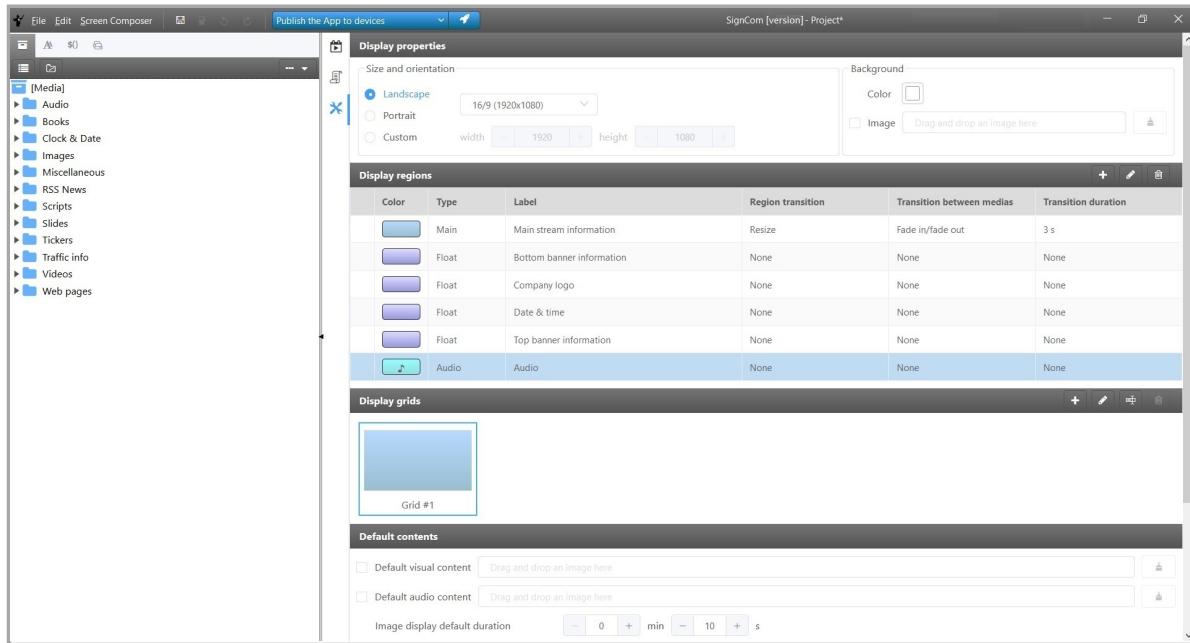
⚠ When the value the layout **(32)** is selected, the column for that display region no longer appears in the media sequence editing window.



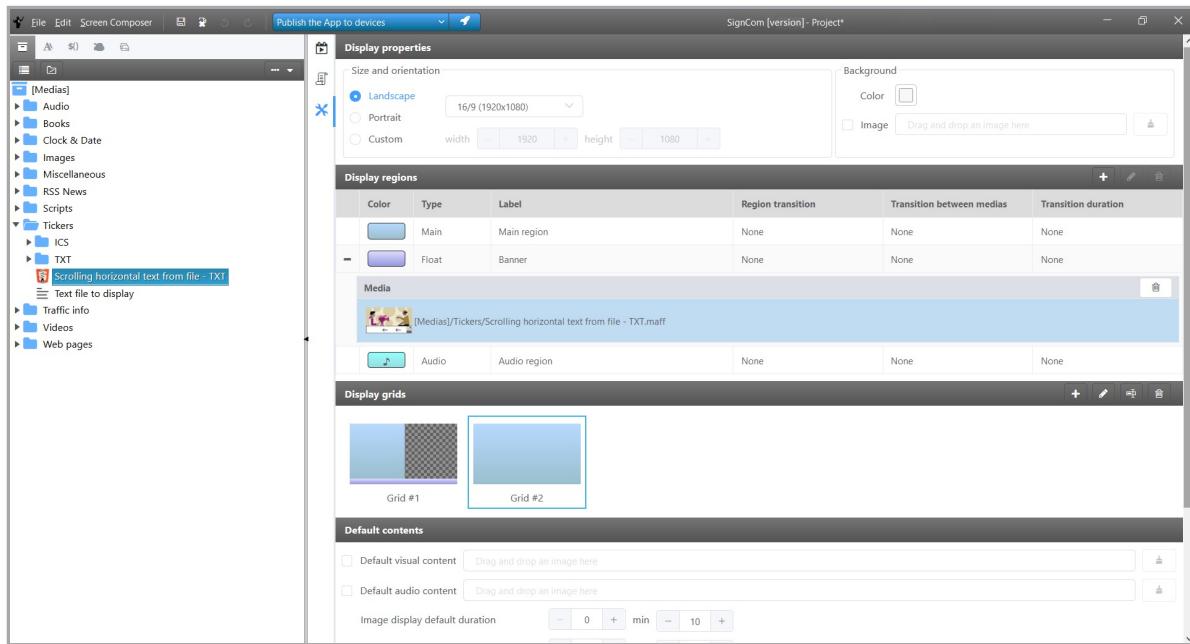
These are the regions created in a summary example with:

- one main display region called *Main stream information*,
- one float display region called *Top banner information*,
- one float display regions called *Bottom banner information*,
- one float display regions called *Date & time*,
- one float display regions called *Company logo*,
- one audio region called *Audio*.

When an *Audio* region is created, the audio track of video played in the visual display grids is muted.



Here's an example of how to assign a single media item to a display area when the value *dressing* (32) is selected for the *Insert media into* attribute of that area:

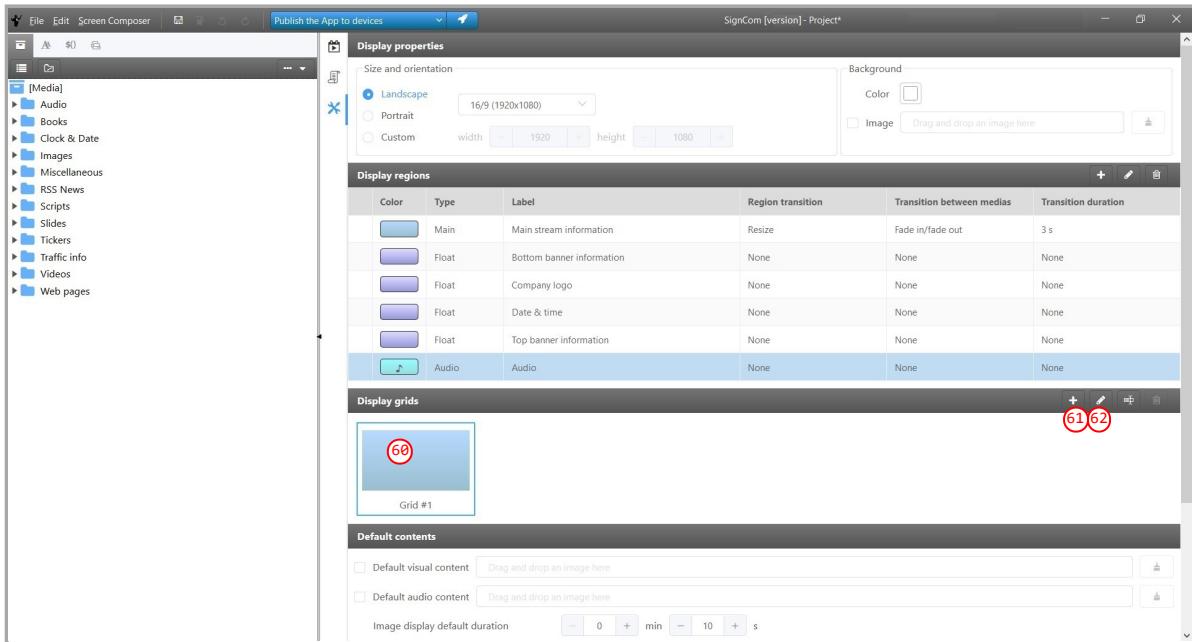


When the value *dressing* (32) is selected, the column for this display area no longer appears in the *media sequences* editing window.

Display grids

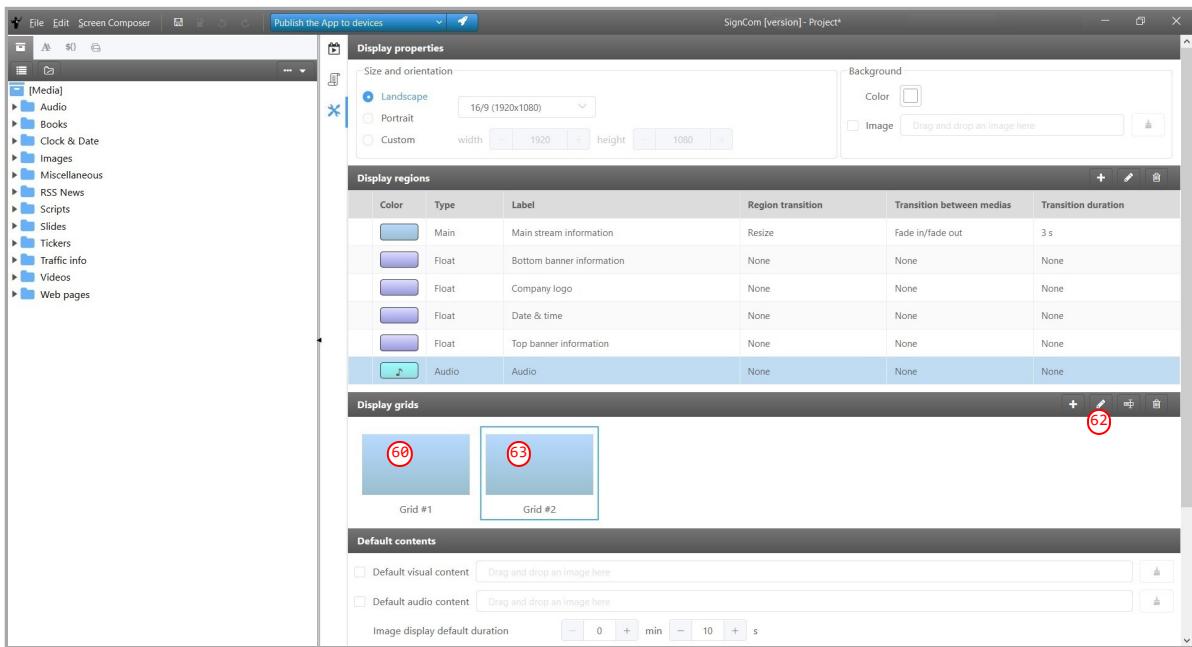
The layout possibilities for your `display grids` are depending on the available `display regions` created.

A default `display region`, filling the full screen (60), represented with the blue color is always available for any new project and for any new created `display grid`.

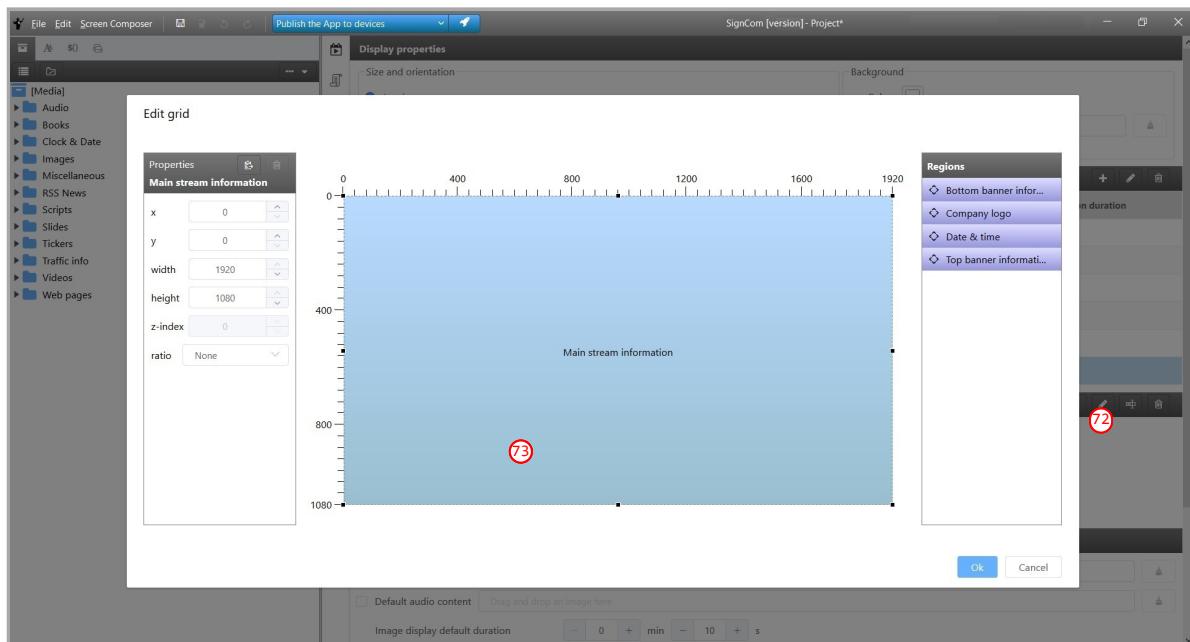


Click on the `Create a grid` + (61) button to create a `display grid`. Enter a label for the `display grid` (by default `Grid #2`).

- ☞ Click on the `Rename` (63) button to rename the grid and watch its unique ID.
- ☞ The `display grids` are displayed here from the left to the right, from the older to the latest created.
- ☞ The `display grid` situated the more at the left (60) (the oldest one) is called the `default display grid`.



Select the appropriate display grid (e.g. Grid #2 ⑥) then click on the Edit  ⑥ button to modify the layout of the display grid.



The main display region (e.g. *Main stream information*) is represented in the default light blue color ⑦⑨ and is by default selected.

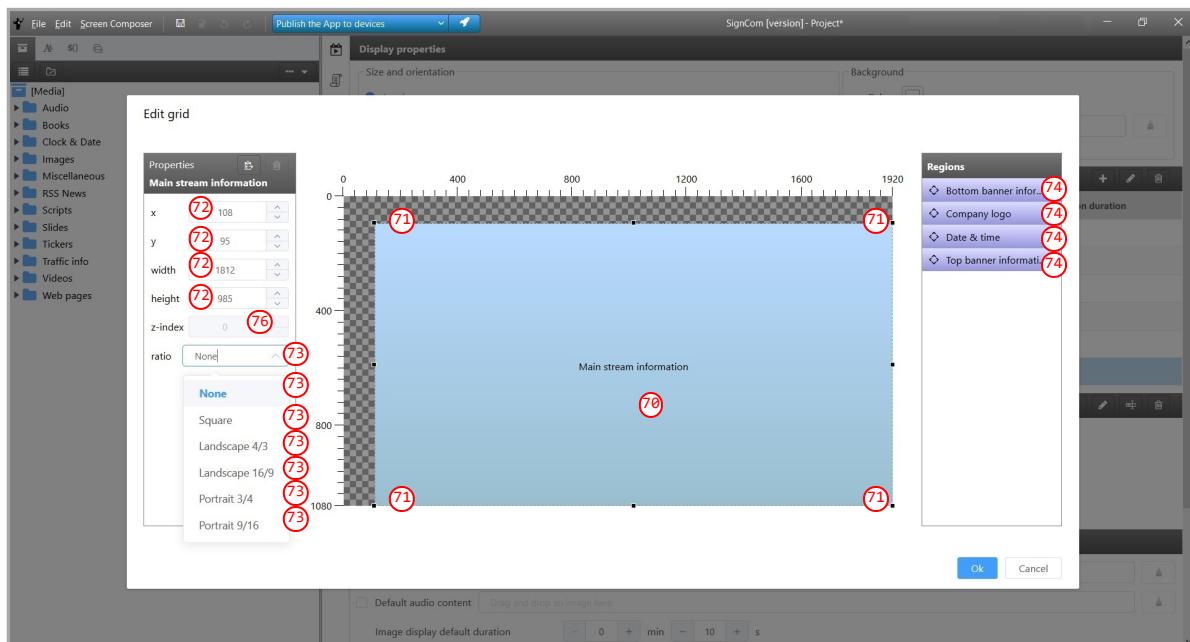
Some handles ⑦⑩ at the edge of the display grid allow to resize it by a simple mouse drag. Helped by the ruler of the layout window, you can do the same by entering the right coordinates ⑦⑪(x, y, width, height) of the display grid in pixels.

It is possible to affect a ratio ⑦⑫ per each display grid.

 The z-index is 0 for the Main display grid.

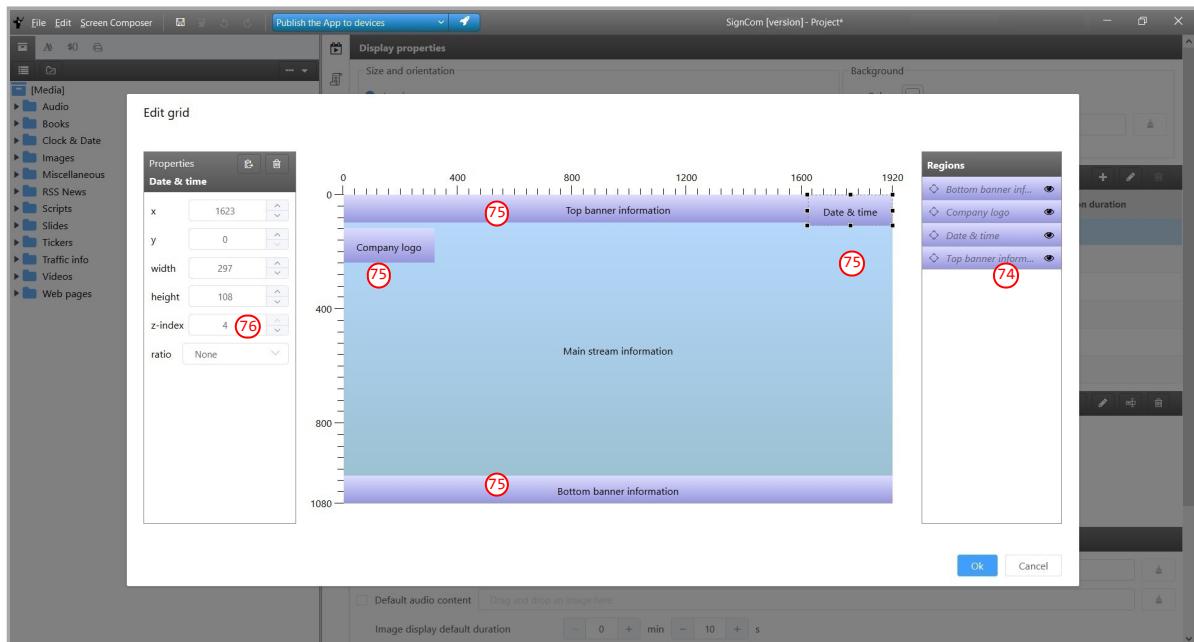
The other available display grids, except the audio region are displayed on the right:

- Bottom banner information (float) ⑦⑭,
- Company logo (float) ⑦⑭,
- Date & time (float) ⑦⑭,
- Top banner information (top banner) ⑦⑭.



Move the wished `display grid` (74) from the right to the layout window (70) by a mouse drag. For `float display grids`, in case overlap, think to set the right the z-index.

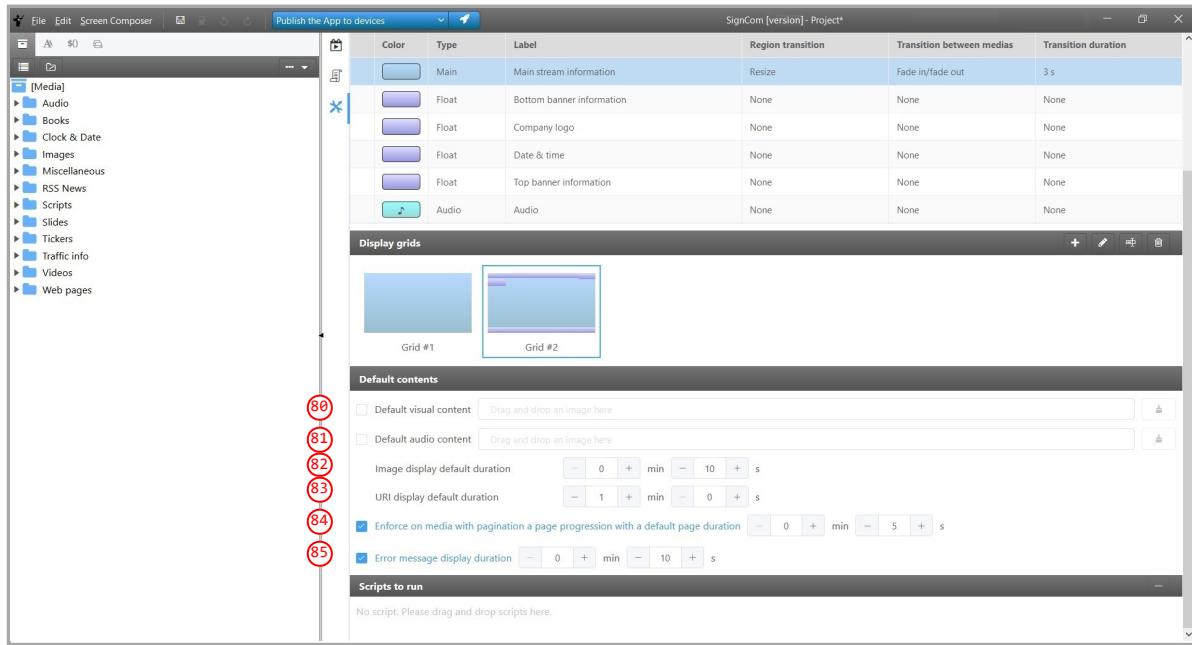
■ Suitable for `float display region` especially, the z-index is an integer value allowing to define the grid priority (kind of calc depth) in case media overlap. The z-index goes from 1 to n for the `float display regions`. In case overlap, the content running in `display regions` having the upper index has the upper priority and is always displayed entirely. Two `float display regions` cannot have the same z-index value in a same `display grid`.



Default contents

This part allows to define a default behaviour for the SignCom App:

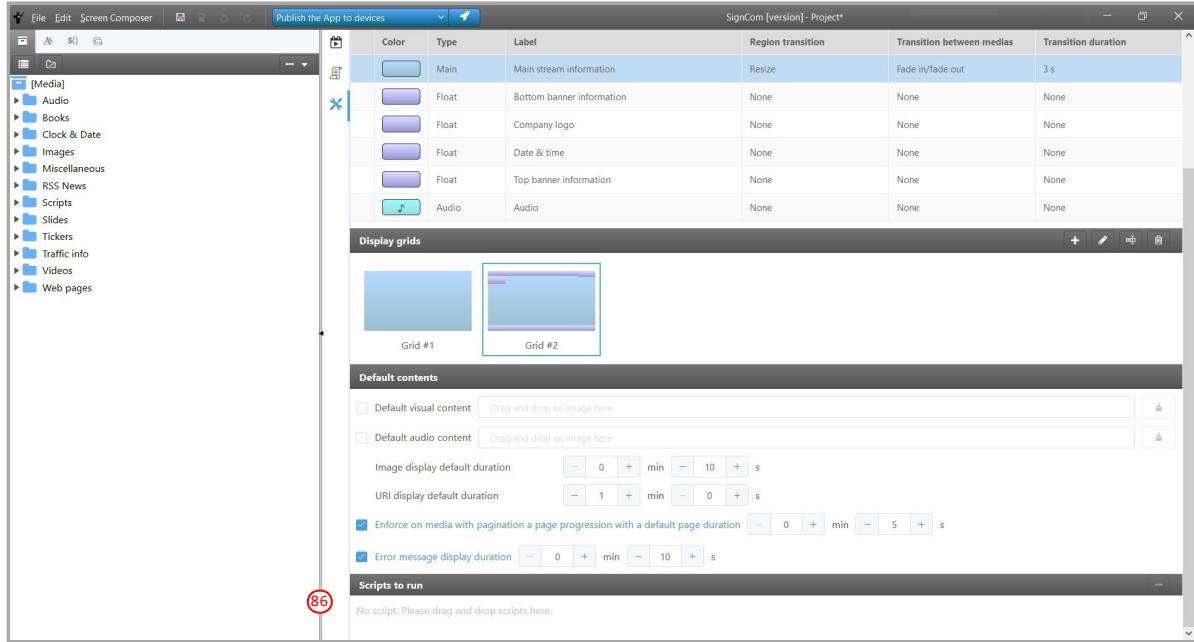
- when no calendar time slot is programmed in the calendar at this time of the day:
 - define a **default visual content** [80](#),
 - define a **default audio content** [81](#),
- for image medias:
 - define a **Image display default duration** [82](#) for images that do not have an already defined duration metadata,
- for URI medias:
 - define a **URI display default duration** [83](#) for URIs that do not have a defined duration metadata,
- PDF and MS-PowerPoint medias:
 - Enforce on media with pagination a page progression with a default page duration to impose for media with pagination a page flow with a default duration of [84](#) n seconds,
- for not consistent media:
 - define an **Error message display duration** [85](#) (for example the information message *content temporarily unavailable code n* is displayed with the specified duration).



⚠ The option **Enforce on media with pagination a page progression with a default page duration** does not work on Gekkota 3 devices.

Script to run

This part allows you to add Javascript file as `scripts to run` (86). These Javascript file can support features, that are not supported natively by the SignCom App, allowing to enhanced the capabilities of the device.



When a script is added to the SignCom App, the Javascript file is executed in parallel of the content to play.

When the device plays another playout with no script to run, the Javascript file is automatically removed from the device after a publication.

⚠️ Javascript files developped for Playzilla / Screen Composer G3 are not compatible with SignCom App.

Part IV

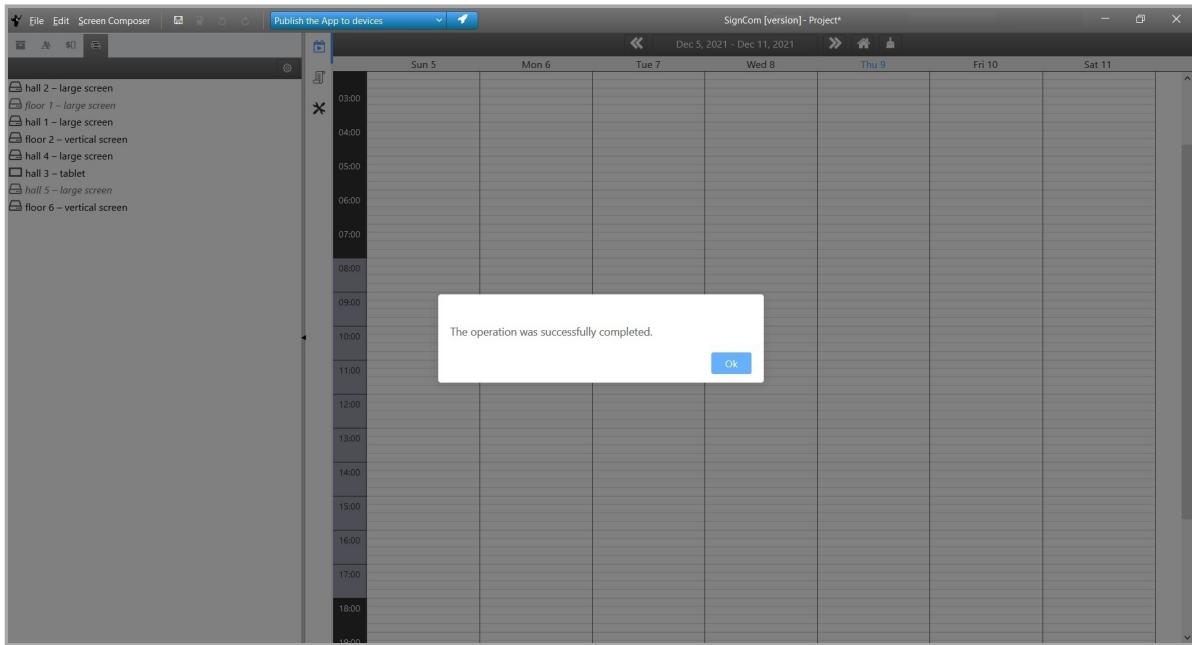
Publication

4.1 App publication

The `Publish to devices` button allow to publish the SignCom App on the devices, applicant for the publication.

There are different ways to publish the App. For further information, refer to the chapter § [File > Project settings > Publication](#).

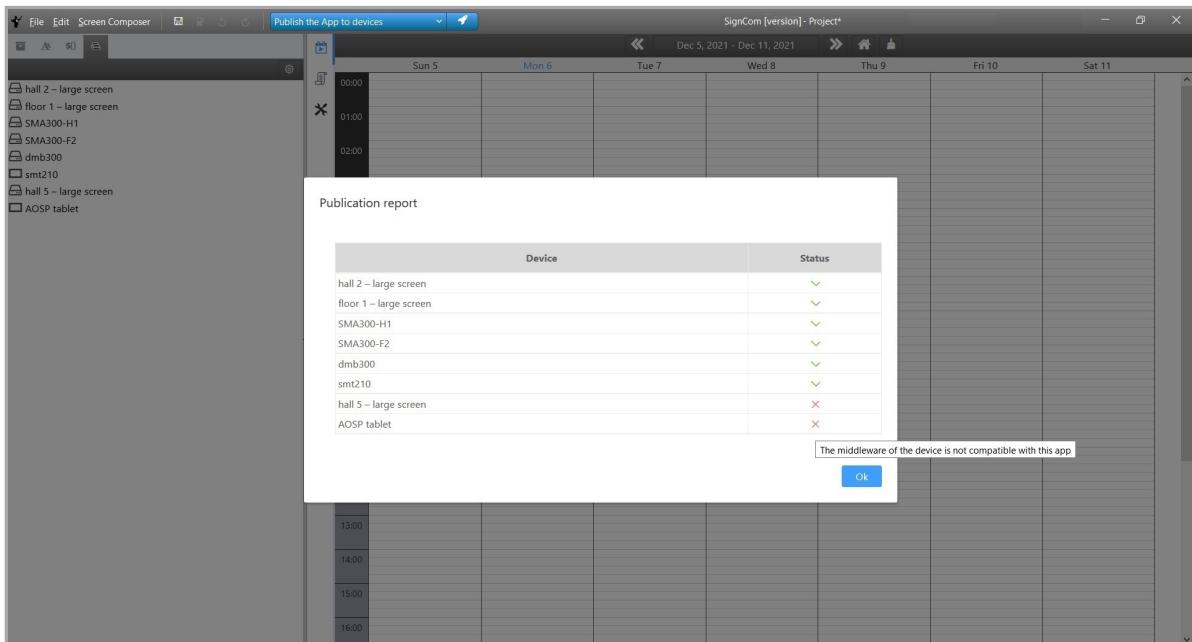
When the publication is successful for all the devices, applicant for publication, the `The operation was successfully completed` message appears.



App publication failure report

When the publication was not successful for at least one device, a `publication report` popup appears after a while and gives information on the publication status.

Publication report status	Information
✓	The SignCom App has been published on the devices.
✗	Screen composer has detected an error in your project. Consequently, the SignCom App cannot be published on some devices.



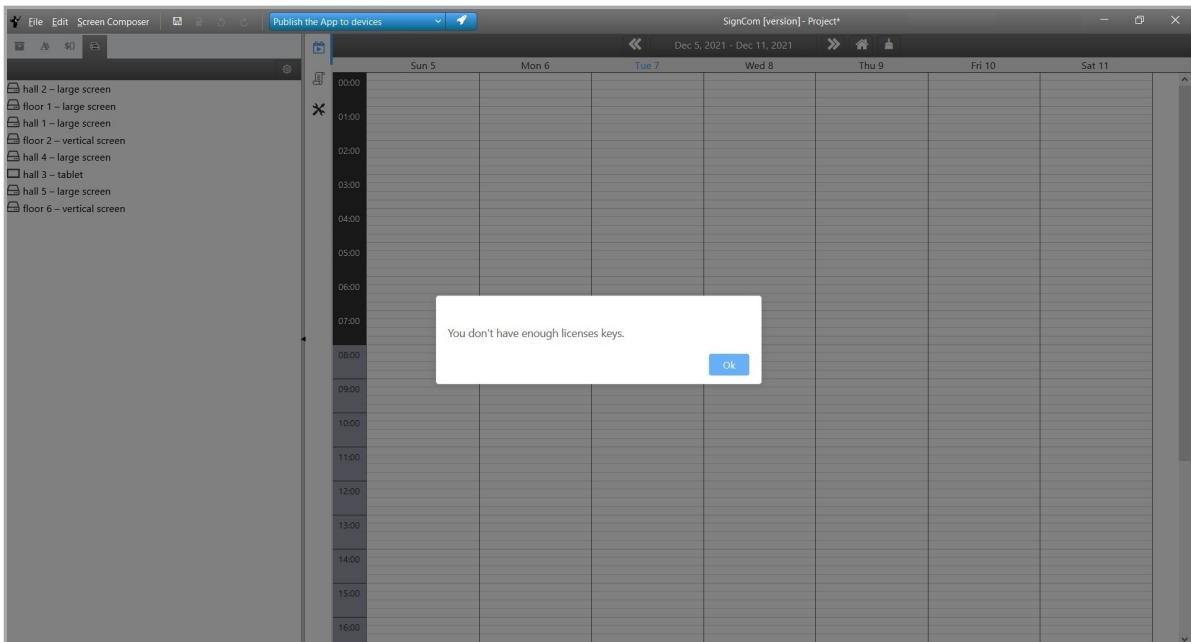
To get more information on the error in your project, fly the mouse pointer over the **✗** red cross to get more information about the publication report status for a device.

The error tooltip is usually very helpful. This table gives more information about how to fix the issue.

Publication report status error tooltip	Information
The middleware of the device is not compatible with this App	Check the compatibility of your SignCom App. Remove from devices applicant for the publication all the ones that are not compatible with this SignCom App (for example, AOSP 9 tablets).
The operation has failed because the device is not accessible	The device is no longer available on the network (it is not supplied, the network cable is removed, the IP address has changed). Check that the device is working properly and retry a device discovery.
Invalid background image file	A background image is required in the form but no image has been chosen.
Invalid default audio content file	A default audio content is required in the form but no audio content has been chosen.
Invalid default visual content file	A default visual content is required in the form but no visual content has been chosen.
The operation was not completed correctly because the login, password or port for one of the devices are not valid	The device has been added to your project with a wrong login credentials or wrong port to access the device WebDAV server
An internal error has occurred. Please try the operation later	One of the reasons could be this one: you are using a https:// scheme which is not yet supported in INNES Screen Composer G4 .

Publication error because insufficient App license keys

If you have not enough SignCom license keys installed of a total of core key license for insufficient device, this error is raised at the publication. To work around, remove some devices from your project or install more SignCom App licenses keys. For further information, refer to the chapter § [Screen Composer > Preferences > Licenses and Apps](#)

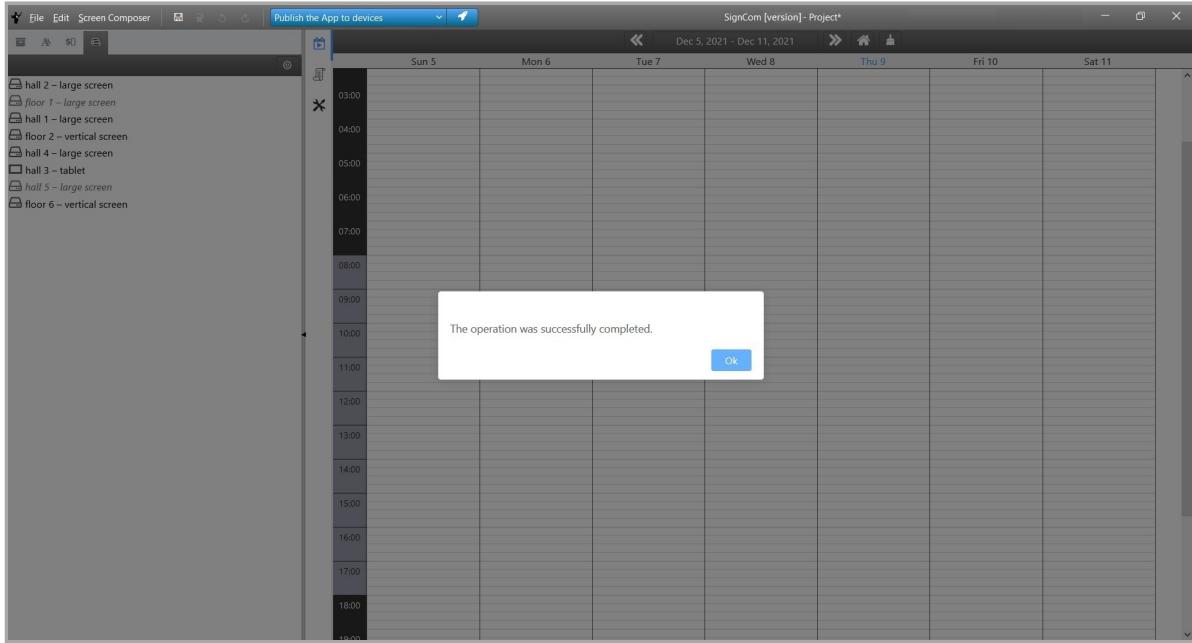


4.2 Middleware publication

When the `Publish middlewares to devices` button value is selected, clicking on the  button allows to publish all the selected middlewares available in the `Screen Composer > Preferences > Middlewares` pane on the devices applicant for the publication.

If the button is not available, refer to the chapter § [File > Project settings > Publication](#) and the chapter § [Screen Composer > Preferences > Middlewares](#).

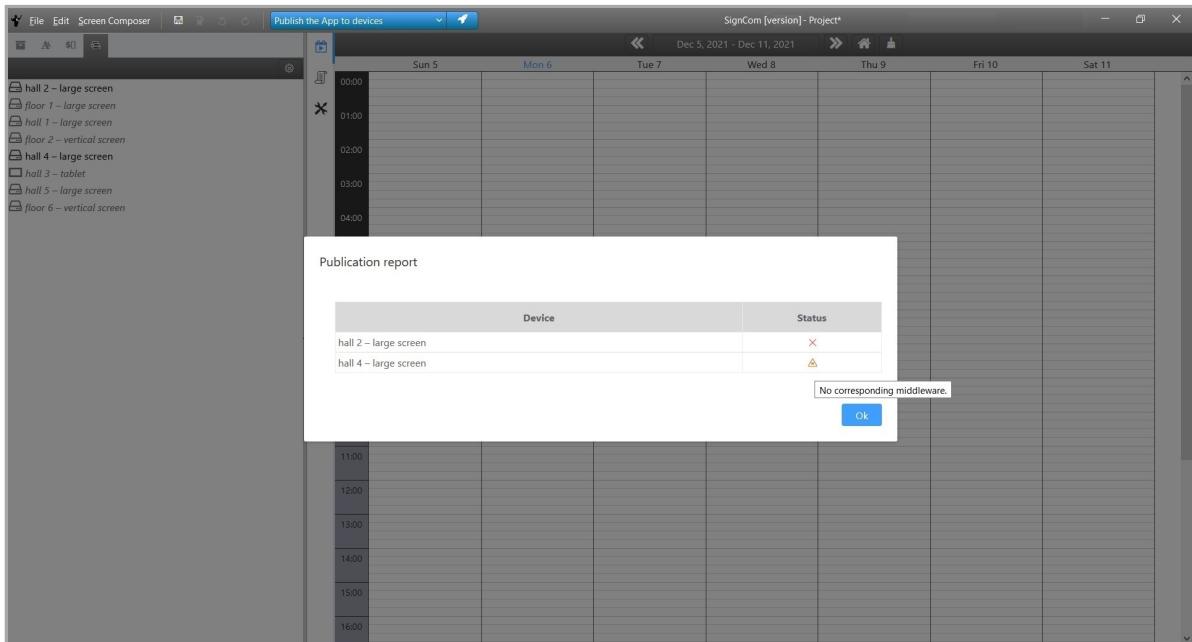
When the publication is successful for all the devices applicant for publication, the information message `The operation was successfully completed` appears.



Middleware publication failure report

When one of the selected devices cannot be updated because there is no suitable middleware for it or the device is not available, a `publication report` popup appears after a while and gives information on the publication status.

Publication report status	Information
	No corresponding middleware.
	The middleware update has failed.



To get more information on the error in your project, fly the mouse pointer over the  red cross to get more information about the publication report status for a device.

The error tooltip is usually very helpful. This table gives more information about how to fix the issue.

Publication report status error tooltip	Information
No corresponding middleware	This device has been selected as applicant for the middlewares publication but there is no appropriate middleware available in the Screen Composer > Preferences > Middlewares pane.
The middleware update has failed	This device is probably not available on the network anymore. So check whether the device is working properly by checking its LED status behaviour. Check the network connectivity between your computer and the device. Check again whether the device is registered in INNES Screen Composer G4 with the appropriate port and IP address value.

Part V

Contacts

5.1 Contacts

For further information, please contact us:

- **Technical support:** support@innes.en,
- **Sales department:** sales@innes.en.

Refer to the INNES Web site for FAQ, application notes, and software downloads: <http://www.innes.pro/en/>.

INNES SA
5A rue Pierre Joseph Colin
35700 RENNES

Tel: +33 (0)2 23 20 01 62
Fax: +33 (0)2 23 20 22 59

Part VI

Appendix

6.1 Appendix: Content model short description

The table below shows a short description of some content models addins.

Content model name	Category	Description	Specific data required to run
<i>Education - Information</i>	Education	Displays information with a specific theme, useful for school establishment.	no
<i>Education - Teachers absence</i>	Education	Displays information about teachers absence, useful for school establishment.	no
<i>Education - Exams</i>	Education	Displays a list of exams, useful for school establishment.	no
<i>Calendar countdown</i>	General	Displays the remaining time (dd:hh:mm:ss) until a next event, recurrent or not.	event date or calendar variable.
<i>News feeds - 16-9</i>	General	Displays the news from your favorite RSS feeds provider.	rss feed URL or .xml file
<i>News feeds - horizontal banner</i>	General	Displays the news from your favorite RSS feeds provider.	rss feed URL or .xml file.
<i>Number of days without accident</i>	General	Displays the number of days elapsed since the last accident happening.	date of the last accident.
<i>Scrolling text from file</i>	General	Displays some scrolling information based on a .txt file.	.txt file having the text information to scroll.
<i>Weather - horizontal banner</i>	General	Displays the weather forecast for the next two days for your favorite country places.	affiliate ID from Alpred SL Web data provider, at least one town name.
<i>World clock</i>	General	Displays from one to three clocks on the same page for maximum three cities.	at least one town name.
<i>Digital clock</i>	General	Displays a digital clock with three themes.	no
<i>MeetingRoom Summarization - Landscape</i>	Signage	Displays agenda of events or of meetings, taking place now or in the future, on one or several pages.	.ics file
<i>MeetingRoom Summarization - Portrait</i>	Signage	Displays agenda of events or of meetings, taking place now or in the future, on one or several pages on portrait screen.	.ics file
<i>MeetingRoom Mono - Landscape</i>	Signage	Displays events taking place today with one event per page.	.ics file
<i>ESII - horizontal banner</i>	Signage	Based on the ESII/eSyrius solution, displays the ticket numbers of queued visitors who have to report to particular counters or services, e.g. town hall services, hospital services, prefecture services, ...	ESII/eSyrius server, port UDP.
<i>Traffic Info Sytadin</i>	Traffic Info France	Display the traffic jam and accidents events on a map of Paris and suburbs.	Sytadin Web data provider.
<i>Festivals and events - tourinsoft - Landscape</i>	Tourism information	Display the incoming festivals and events for a given region in landscape mode.	TourInSoft Web data provider.
<i>Festivals and events - tourinsoft - Portrait</i>	Tourism information	Display the incoming festivals and events for a given region in portrait mode.	TourInSoft Web data provider.

6.2 Appendix: Medias ending management for HTML widgets and URI based on models

Depending on the previewed media, the duration of the entire media preview can be more and less long. For some of them like URIs, it never ends. That is the same once published on the device. This table explains the good practise on how to ensure that a media is endind after a while.

In case a media has an intrinsic duration, that means that is is ending naturally after a while.

Medias type	Intrinsic duration	Medias ending management
- Digital clock HTML widget, - Education - Information HTML widget, - Education - Teachers absence HTML widget, - Education - Exams HTML widget, - Number of days without accident HTML widget, - World clock HTML widget, - Web TV URI , - HDMI-input URI , - DVB-T URI	no	- by adding a value in second for the duration metadata.
- Calendar countdown HTML widget	no	- is depending on the current date and time or - by adding a value in second for the duration metadata.
- Weather horizontal banner HTML widget, - Traffic Info Sytadin HTML widget, - MeetingRoom - Mono HTML widget, - MeetingRoom - Landscape HTML widget, - RSS Wide HTML widget, - News feeds - horizontal banner HTML widget - Widget HTML Festivals and events - tourinsoft	no	- is depending on the amount of data provided by the Web service data or - by adding a value in second for the duration metadata.
- Scrolling text from file HTML widget	no	- is depending on amount of lines in the text file or - by adding a value in second for the duration metadata.

6.3 Appendix: supported inputs and buttons properties to support identifier and password self-filling and auto-confirmation in a Web page form

List of supported input properties in the Web page form to support the *identifier* field self-filling

email type input

*user i name** input

email autocomplete input

*user i id** input

*login i id** input

*email i id** input

List of supported input properties in the Web page form to support the *password* field self-filling

password input type

List of supported validate button properties in the Web page form to support the auto-confirmation

submit type input

submit type button

button type input

*sign id** input

*submit id** input

6.4 Appendix: Azure AD portal

You can create your Azure Active Directory (or AAD) application by following this Microsoft tutorial <https://docs.microsoft.com/en-us/graph/auth-register-app-v2>.

A procedure example is shown here after by connecting to the Microsoft Azure portal.

This procedure allows to generate you own ID and SECRET required in the SignCom App:

- Application (client) ID ,
- Directory (Tenant) ID ,
- Client secret .

If you want to follow the PowerShell scripts procedure instead of following the procedure by connecting to the Azure AD portal, only PowerShell script for Azure Active Directory Application support 1.10.13 (and above) is supported. For further information, refer to the chapter § Appendix: Azure AD Application PowerShell module.

Connect on Microsoft Azure portal: <https://portal.azure.com/> and sign in with your Office Administrator account login credentials.

Click on the left top menu and choose the Azure Active directory item.

The screenshot shows the Microsoft Azure portal home page. At the top, there's a search bar and a navigation bar with icons for Home, Dashboards, All services, Favorites, and more. Below the search bar, there are three main promotional cards: "Start with an Azure free trial", "Manage Azure Active Directory", and "Access student benefits". Under "Azure services", there's a row of icons for Create a resource, Azure Active Directory, App Service Domains, Azure AD B2C, Subscriptions, AD Connect, App Services, All resources, Function App, and More services. At the bottom, there's a "Navigate" section with links for Home, Dashboard, All services, Favorites, and other Azure services like App Services, Function App, and More services.

This screenshot is identical to the one above, but the "Azure Active Directory" link in the left sidebar under the "Favorites" section is highlighted with a red box. The rest of the interface is the same, showing the home page with its various promotional cards and service icons.

Application (client) ID and directory (Tenant) ID

On the App registrations menu, click on the *New registration* button.

https://portal.azure.com/#blade/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/RegisteredApps.

The screenshot shows the 'App registrations' blade in the Azure portal. On the left, a sidebar lists various management options like Overview, Preview features, and Diagnose and solve problems. Under the 'Manage' section, 'App registrations' is selected and highlighted with an orange box. At the top of the main content area, there's a toolbar with buttons for 'New registration' (which is also highlighted with an orange box), Endpoints, Troubleshooting, Refresh, Download, Preview features, and Got feedback?.

Enter an application name (e.g.: *playfolder-with-microsoft-365-application*), Select the appropriate Account in the organisation directory only (organisation only – Single tenant) radio button, and click on the Register button.

The screenshot shows the 'Register an application' blade. The 'Name' field contains the value 'playfolders-and-uris-with-microsoft-365-services-account'. Below the name field, under 'Supported account types', the radio button for 'Accounts in this organizational directory only (INNES only - Single tenant)' is selected. Other options include 'Accounts in any organizational directory (Any Azure AD directory - Multitenant)', 'Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)', and 'Personal Microsoft accounts only'. There is also a 'Help me choose...' link. Further down, there's a 'Redirect URI (optional)' section with a dropdown for 'Select a platform' and a text input field containing 'e.g. https://example.com/auth'. A note states: 'We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.' At the bottom, there's a link 'By proceeding, you agree to the Microsoft Platform Policies' and a prominent 'Register' button which is highlighted with an orange box.

In the Overview menu, copy to clipboard the Application (client) ID value, the 1st value required in SignCom App configuration tab and store it previously.

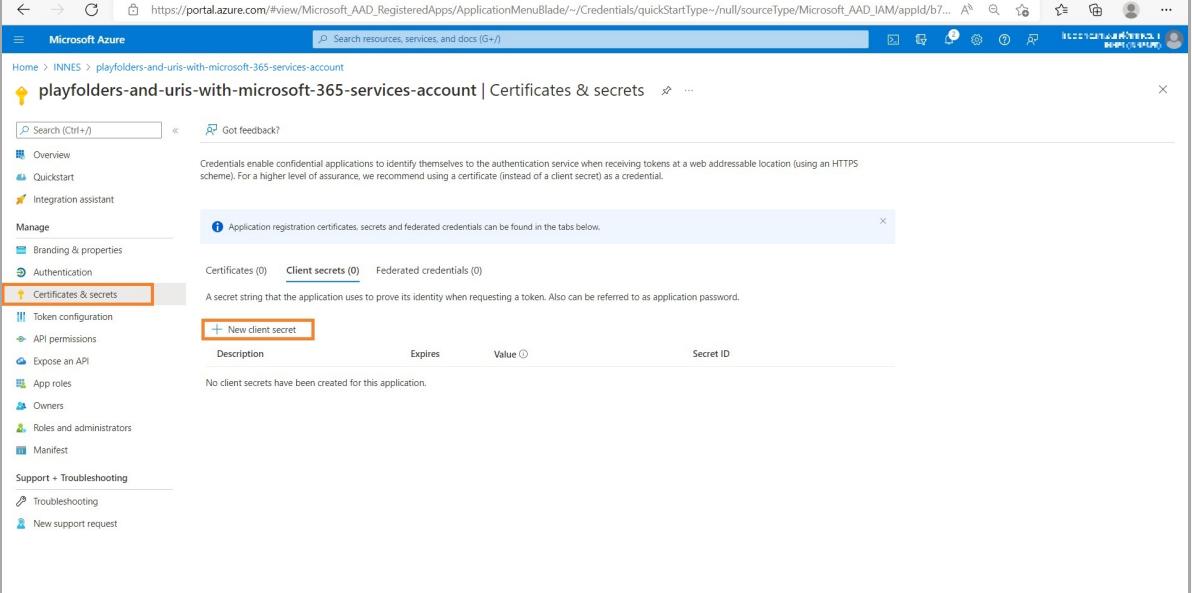
The screenshot shows the Microsoft Azure portal with the URL https://portal.azure.com/#view/Microsoft_AAD_RegisteredApps/ApplicationMenuBlade/~/Overview/quickStartType~/null/sourceType/Microsoft_AAD_IAM/appId/b75a.... The page title is "playfolders-and-uris-with-microsoft-365-services-account". The left sidebar includes links for Overview, Quickstart, Integration assistant, Manage, Authentication, Certificates & secrets, Token configuration, API permissions, Expose an API, App roles, Owners, Roles and administrators, and Manifest. The main content area is titled "Essentials" and shows fields for Display name (playfolders-and-uris-with-microsoft), Application (client) ID (highlighted with a red box), Object ID, Directory (tenant) ID (highlighted with a red box), and Supported account types (My organization only). A "Copy to clipboard" button is visible next to the Application (client) ID and Directory (tenant) ID fields. Below this, there are two informational cards: one about the new improved App registrations and another about the deprecation of Azure Active Directory Authentication Library (ADAL) and Azure AD Graph starting June 30th, 2020. At the bottom, there are "Get Started" and "Documentation" links, and a "Build your application with the Microsoft identity platform" section.

In the Overview menu, copy to clipboard the Directory (tenant) ID value, the 2nd value required in SignCom App configuration tab and store it preciously.

This screenshot is identical to the one above, showing the Microsoft Azure portal Overview page for the same app registration. The "Directory (tenant) ID" field is highlighted with a red box, and a "Copy to clipboard" button is visible next to it. The rest of the interface, including the sidebar, essential information, and informational cards, is the same.

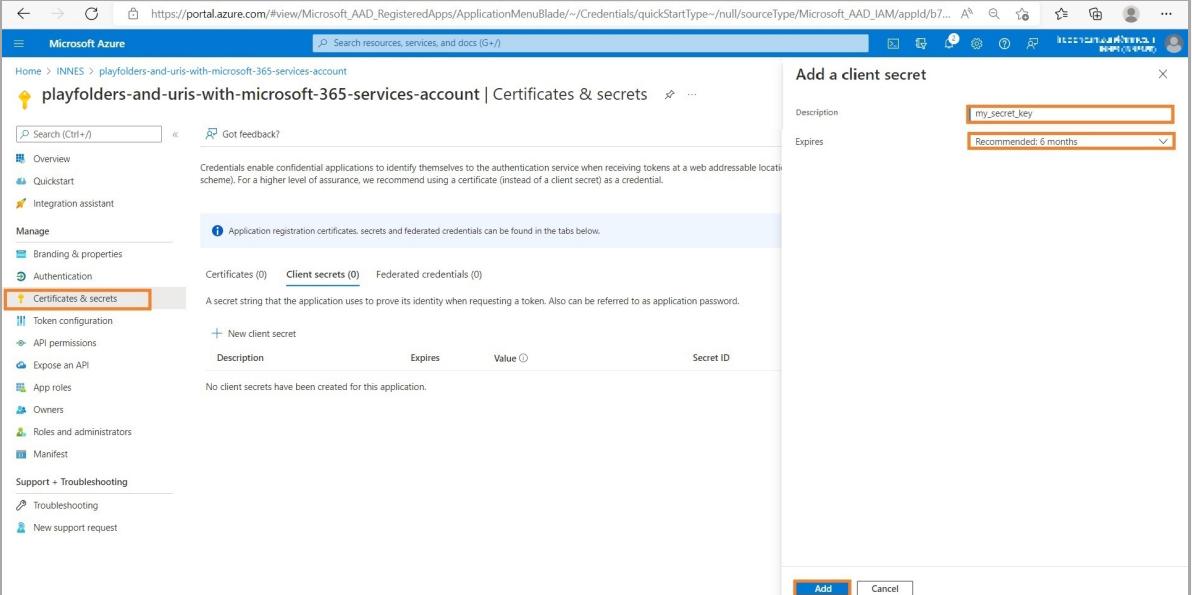
Client secret

In the Certificates & secrets menu, click on the New client secret button.



The screenshot shows the Microsoft Azure portal interface for managing application registration. The left sidebar has 'Certificates & secrets' selected. In the main content area, the 'Client secrets' tab is active. A button labeled '+ New client secret' is highlighted with a red box. Below it, there are columns for Description, Expires, Value (which is currently empty), and Secret ID.

Enter a name (e.g.: `my_secret_key`) and click on the Add button.



The screenshot shows the 'Add a client secret' dialog box overlaid on the Azure portal. It has fields for 'Description' (containing 'my_secret_key') and 'Expires' (set to 'Recommended: 6 months'). At the bottom right are 'Add' and 'Cancel' buttons, with 'Add' highlighted with a red box.

Copy into clip board the client secret value, the 3rd input for the SignCom App configuration tab and store it preciously.

⚠ Do it right now because the client secret value is not visible anymore as soon as you click on a new Web page.

The screenshot shows the Microsoft Azure portal interface. The URL in the address bar is https://portal.azure.com/#view/Microsoft_AAD_RegisteredApps/ApplicationMenuBlade/~/Credentials/quickStartType~/null/sourceType/Microsoft_AAD_IAM/appId/b7.... The page title is "playfolders-and-uris-with-microsoft-365-services-account | Certificates & secrets".

The left sidebar navigation includes:

- Overview
- Quickstart
- Integration assistant
- Manage
 - Branding & properties
 - Authentication
 - Certificates & secrets**
 - Token configuration
 - API permissions
 - Expose an API
 - App roles
 - Owners
 - Roles and administrators
 - Manifest
- Support + Troubleshooting
 - Troubleshooting
 - New support request

The main content area displays the "Certificates & secrets" tab, which is currently selected. It shows a table with one row:

Description	Expires	Value	Copy to clipboard	Get ID
my_secret_key	12/3/2022	UoeBQ~8hr-A-BoQv5B055bwV48GnzK. 7dde83bb-3aa1-467e-a4e9-d7527b348d...	<input type="button" value="Copy to clipboard"/>	<input type="button" value="Get ID"/>

A tooltip message above the table reads: "Application registration certificates, secrets and federated credentials can be found in the tabs below."

A feedback dialog box at the top right says: "Got a second to give us some feedback? →".

Grant permissions

For the `playfolders-and-uris` application, these permissions must be granted:

- Application permissions:
 - `Files.Read.All`,
 - `Sites.Read.All`,
- Delegated permissions:
 - `Files.Read.All`,
 - `Sites.Read.All`,
 - `User.Read` (default).

Note The `Files.Read.All` permission allows to read files or folder content on oneDrive. The `Sites.Read.ALL` permission allows to read the Web URL.

In the API permissions menu, click on the Add a permission button.

Select Microsoft Graph button in the Microsoft APIs tab.

Application permission:

Select then the Application permissions button.

The screenshot shows the Microsoft Azure portal with the URL https://portal.azure.com/#view/Microsoft_AAD_RegisteredApps/ApplicationMenuBlade/~/~/CallAnAPI/appId/b75a90a8-ba42-4969-8622-532be827402e/isMSAApp~/fa.... The left sidebar shows the 'Manage' section with 'API permissions' selected. The main area displays the 'Request API permissions' dialog for Microsoft Graph. The 'Delegated permissions' section shows a single permission: 'User.Read' (Type: Delegated, Description: Sign in and read user profile). The 'Application permissions' section is collapsed. At the bottom right of the dialog are 'Add permissions' and 'Discard' buttons.

In the display filter input, enter the text `Files` and check the option `Files.Read.All`.

Do not click now on the `Add permissions` button.

The screenshot shows the Microsoft Azure portal with the same URL as the previous screenshot. The 'Select permissions' dialog is open, showing the 'Files' category expanded. Under 'Files', the 'Files.Read.All' permission is selected, indicated by a checked checkbox and highlighted with a red border. Other options like 'Files.ReadWrite.All' and 'OnPremisesPublishingProfiles' are also listed. The 'Add permissions' button at the bottom right is highlighted with a red border.

To be able to read Sharepoint Web URL or Sharepoint Web sites, in the display filter input, enter the text **Sites** then check the option **Sites.Read.All**.

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various options like Overview, Quickstart, Integration assistant, Manage, API permissions (which is selected), Expose an API, App roles, Owners, Roles and administrators, and Manifest. The main area is titled 'Request API permissions' for a Microsoft Graph application. It shows a warning about admin consent required. Under 'Delegated permissions', there's a section for 'Application permissions'. A table lists permissions under 'Select permissions':

API / Permissions name	Type	Description	Admin consent required
Sites	Application	Read and write files in all site collections	Yes
Sites (1)			
<input type="checkbox"/> Sites.FullControlAll	<input type="radio"/>	Have full control of all site collections	Yes
<input type="checkbox"/> Sites.Manage.All	<input type="radio"/>	Create, edit, and delete items and lists in all site collections	Yes
<input checked="" type="checkbox"/> Sites.Read.All	<input type="radio"/>	Read items in all site collections	Yes
<input type="checkbox"/> Sites.ReadWrite.All	<input type="radio"/>	Read and write items in all site collections	Yes
<input type="checkbox"/> Sites.Selected	<input type="radio"/>	Access selected site collections	Yes

At the bottom right of the dialog are 'Add permissions' and 'Discard' buttons.

Click on the **Add permissions** button.

Delegated permissions:

Select then the **Delegated permissions** button.

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various options like Overview, Quickstart, Integration assistant, Manage, and API permissions. The API permissions section is currently selected. On the right, a modal window titled "Request API permissions" is open, specifically for "Microsoft Graph". Inside, there are two main sections: "Delegated permissions" (which is highlighted with an orange box) and "Application permissions". The "Delegated permissions" section contains a table with five rows of permissions, all of which are currently selected (indicated by blue checkboxes). The "Application permissions" section is also visible but contains no selected items. At the bottom of the modal, there are "Add permissions" and "Discard" buttons.

API / Permissions name	Type	Description
Files.ReadWrite.All	Delegated	Have full access to all files user can access
Files.ReadWrite.All	Application	Read and write files in all site collections
Sites.ReadWrite.All	Application	Read and write items in all site collections
User.Read	Delegated	Sign in and read user profile
User.Read.All	Application	Read all users' full profiles

In the display filter input, enter the text `Files` and check the option `Files.Read.All`.

Do not click now on the `Add permissions` button.

This screenshot shows the same "Request API permissions" dialog for Microsoft Graph, but with a different focus. The "Select permissions" section is now active, showing a search bar with the text "Files" and a list of permissions under the "Files" category. One permission, "Files.Read.All", has a checked checkbox, which is highlighted with an orange box. The other permissions in the list are not checked. The rest of the dialog and sidebar are identical to the previous screenshot.

Permission	Admin consent required
CrossTenantUserProfileSharing	No
Files (1)	
<input type="checkbox"/> Files.Read (1) Read user files	No
<input checked="" type="checkbox"/> Files.Read.All (1) Read all files that user can access	No

To be able to read Sharepoint Web URL or Sharepoint Web sites, in the display filter input, enter the text **Sites** then check the option **Sites.Read.All**.

The screenshot shows the 'Request API permissions' dialog in the Azure portal. The 'Delegated permissions' section is highlighted. A permission named 'Sites.Read.All' is selected, indicated by a checked checkbox. The 'Admin consent required' column for this permission shows 'No'. Other permissions listed include 'Files.ReadWrite.All', 'Files.ReadWrite.All', 'Sites.ReadWrite.All', 'User.Read', and 'User.Read.All', all with 'Yes' in the 'Admin consent required' column.

At this step, the permissions are not yet granted.

Grant permissions:

Click on the `Grant admin consent for <your_organisation>` button .

The screenshot shows the 'Grant admin consent for INNES' button being clicked. A success message 'Successfully saved permissions for playfolders-and-uris-with-microsoft-365-services-account.' is displayed in a toast notification. The main table now shows the status for each permission: 'Sites.Read.All' is marked with a warning icon and 'Not granted for INNES', while other permissions like 'Files.Read.All' and 'User.Read' have 'No' in the 'Status' column.

Grant admin consent confirmation.

Do you want to grant consent for the requested permissions for all accounts in INNES? This will update any existing admin consent records this application already has to match what is listed below.

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

API / Permissions name	Type	Description	Admin consent requ...	Status	...
Microsoft Graph (5)					
Files.Read.All	Delegated	Read all files that user can access	No		...
Files.Read.All	Application	Read files in all site collections	Yes	⚠️ Not granted for INNES	...
Sites.Read.All	Delegated	Read items in all site collections	No		...
Sites.Read.All	Application	Read items in all site collections	Yes	⚠️ Not granted for INNES	...
User.Read	Delegated	Sign in and read user profile	No		...

To view and manage permissions and user consent, try [Enterprise applications](#).

Now the permissions are granted.

Successfully granted admin consent for the requested permissions.

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value in your organization, or in organizations where this app will be used. [Learn more](#)

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

API / Permissions name	Type	Description	Admin consent requ...	Status	...
Microsoft Graph (5)					
Files.Read.All	Delegated	Read all files that user can access	No	Granted for INNES	...
Files.Read.All	Application	Read files in all site collections	Yes	Granted for INNES	...
Sites.Read.All	Delegated	Read items in all site collections	No	Granted for INNES	...
Sites.Read.All	Application	Read items in all site collections	Yes	Granted for INNES	...
User.Read	Delegated	Sign in and read user profile	No	Granted for INNES	...

To view and manage permissions and user consent, try [Enterprise applications](#).

6.5 Appendix: Azure AD Application PowerShell module

This is the suitable PowerShell module to create your Azure Active Directory Application for playfolders or uris based on a Microsoft 365 services account .

⚠️ For playfolders and uris application, the PowerShell script for Azure Active Directory Application support (`Powershell_Innes_AAD`) must be 1.10.16 (or above).

Download the PowerShell script for Azure Active Directory Application support `Powershell_Innes_AAD-1.10.16.zip` from the [Innes Site Web](#) then follow the instructions below.

Compatibility

The `Powershell_Innes_AAD-1.10.16.zip` PowerShell script for Azure Active Directory application is compatible with `PowerShell 5.X` (deployed on Windows 10).

Introduction

This set of `Powershell` functions allows to:

- create an *Azure Active Directory* application, with the `New-AADApplication` function,
- remove an *Azure Active Directory* application, with the `Remove-AADApplication` function.

These functions are defined in the `PSAAD` PowerShell module stored in the `Modules\PSAAD\` directory.

The result of the `Powershell` functions is also stored in a JSON file.

Edit the file and store preciously the values which could be required for your application:

- the `clientId` value,
- the `tenantId` value,
- the `clientSecret` value.

Security

By default, the execution of local `Powershell` scripts are not allowed. You can change their execution rights by changing the `PowerShell` security policy. This modification has to be done once with the `Set-ExecutionPolicy` `Powershell` function. Your organisation may have to change it according to your security rules.

For example, to authorize the execution of all scripts, launch a `Powershell` console with administrator rights, and type:

```
PS > Set-ExecutionPolicy -ExecutionPolicy Unrestricted -scope CurrentUser
```

For further information, look at the cmdlet `Set-ExecutionPolicy` help page.

If you cannot allow the execution of unsigned local scripts, you can install the provided certificate in the list of authorized root certificates with the command:

```
PS > cd <your_path_to_the_scripts>\Powershell_Innes_AAD\Certificate\  
PS > Import-PfxCertificate -FilePath InnesCodeSigningRootCA_1.pfx -CertStoreLocation .../  
cert:\CurrentUser\Root -Password $(ConvertTo-SecureString "1234" -AsPlainText -Force)
```

To import the `.pfx` certificate, you can also use the MS-Windows application `certmgr.msc`, select the *Trusted Root Certification Authorities*, right click on *All Tasks*, select the *Import* item, select the file and enter the password `1234`. When ended, close the current `Powershell` console.

Prerequisite

Install the AzureAD module

Install the `AzureAD` module with the command below:

```
PS > Install-Module -name AzureAD -scope CurrentUser
```

Dependency

If this message is prompted, enter `y`.

```
The NuGet supplier is required to continue  
PowerShellGet requires the NuGet vendor, version 2.8.5.201 or later, to interact with the repositories.  
The NuGet provider must be available in "C:\Program Files\PackageManagement\ProviderAssemblies" or .../  
"C:\Users\<username>\AppData\Local\PackageManagement\ProviderAssemblies".  
You can also install the provider NuGet by executing the command "Install-PackageProvider -Name NuGet .../  
-MinimumVersion 2.8.5.201 -Force". Do you want that PowerShellGet installs and imports the NuGet provider now?  
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"):
```

If this message is prompted, enter `y`.

```
Unapproved repository
You install the modules from an unapproved repository. If you approve this repository, .../
change its InstallationPolicy value by running the Set-PSRepository command applet. .../
Do you really want to install From PSGallery ?
[Y] Yes [T] Yes for all [N] No [U] No for all [S] Suspend [?] Help (default is "N"):
```

Usage

To use one of the *Powershell* modules, you have to define the environment variable for PSAAD. You have 3 possibilities:

1. Either copy the directories under `Modules\` into a standard *Powershell* module installation directory, for example `C:\Program Files\WindowsPowerShell\Modules`. Then launch a *Powershell* console.
2. Or redefine the search variable for *Powershell* modules (the `$Env:PSModulePath Powershell` variable) each time you will use theses functions. In this case, launch a *Powershell* console, and type the line below, adapting it to your path. Each time you launch a new *Powershell* console, you need to enter it again.

Example:

```
PS > $Env:PSModulePath="$Env:PSModulePath;C:\Program Files (x86)\WindowsPowerShell\Modules"
```

3. Or redefine the search variable for *Powershell* modules in the Windows environment variables. For that, add the path `<your_path_to_the_scripts>\Powershell_Innes_AAD\Modules` to the environment variable `PSModulePath`. Then, launch afterwards a *Powershell* console.

To use the functions or get help, you must then import the module(s) with the `Import-Module` function. Example:

```
PS > Import-Module PSAAD
```

Depending on how you get the scripts, you may have this following warning:

```
Security Warning Run only scripts that you trust. While scripts from the Internet can be useful, .../
this script can potentially harm your computer. Do you want to run \server\scripts\my.ps1? .../
[D] Do not run [R] Run once [S] Suspend [?] Help (default is "D"):
```

To avoid this message, you can unblock the script files (to do only once):

```
PS > cd <your_path_to_the_scripts>\Powershell_Innes_AAD\
PS > dir -Recurse | Unblock-File
```

The `Get-Command` function allows you to list the functions defined in a module. Example:

```
PS > Get-Command -Module PSAAD
```

Answer example:

CommandType	Name	Version	Source
Function	New-AADApplication	1.10.16	PSAAD
Function	Remove-AADApplication	1.10.16	PSAAD

You can get help on each function of the module by using the standard cmdlet `Get-Help` with options:

- `-detailed`,
- `-full`,
- `-examples`.

Example:

```
PS > Get-Help -detailed New-AADApplication
```

NAME
New-AADApplication

SYNOPSIS
This function creates a Azure Active Directory application.

SYNTAX
New-AADApplication [[-Credential] <PSCredential>] [[-tenantId] <String>] [-appName] <String> [-authorizations] <String[]> [[-LogFile] <String>] [<CommonParameters>]

DESCRIPTION
This function creates a Azure Active Directory application.

PARAMETERS
-Credential <PSCredential>
 Credential (admin profile) used to create the Azure Active Directory application. If absent, a dialog is displayed in the browser to enter the credentials.

-tenantId <String>
 Azure Active Directory Tenant Id of the tenant in which the application has been created. This parameter is not mandatory. If absent, the tenantId is retrieved automatically after the credentials have been entered in the dialog.

-appName <String>
 Name of the Azure Active Directory application.

-authorizations <String[]>
 Authorization type:
 - "signcom_m365" : to access to M365 files and folders resources and Web sites for SignCom application
 - "url_launcher_m365" : to access to M365 Web sites for URL launcher application
 - "signmeeting_ews": to access to MS-Exchange room mailbox resources for SignMeeting MS-Exchange application
 - "signmeeting_m365": to access to M365 room mailbox resources for SignMeeting-M365 application
 - "briva_calendar_ews": to access to MS-Exchange room mailbox resources for Briva Calendar EWS application
 - "m365_room": to access to M365 room mailbox resource for SBL10e m365_room application
 - "m365_user": to access to M365 user presence resource for SBL10e m365_user application
 - "powerbi": to access to Power BI report

-LogFile <String>
 Log file path

<CommonParameters>
 This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see about_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

----- EXAMPLE 1 -----

PS C:\>\$result = New-AADApplication -appname "my-App-Label" -authorizations "Authorization type"

A consent request will be sent in 30 seconds in your browser.
You must log into an administrator account of your organization and grant the necessary permissions.

PS C:\>\$result

Name	Value
clientId	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
objectId	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
spId	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
name	my-App-Label
tenantId	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
clientSecret	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

REMARKS

To see the examples, type: "get-help New-AADApplication -examples".
For more information, type: "get-help New-AADApplication -detailed".
For technical information, type: "get-help New-AADApplication -full".

Example to create an Azure Active Directory application

```
PS > $result = New-AADApplication -appname "playfolders-and-uris-with-microsoft-365-service-account" -authorizations "signcom_m365"
```

- ☞ Don't use space characters inside the appname else an error could be returned.
- ☞ Don't use an already existing Appname else an error is returned.
- ⚠ Clicking on a Powershell window can suspend the command. In this case click again in the window to resume the command.*

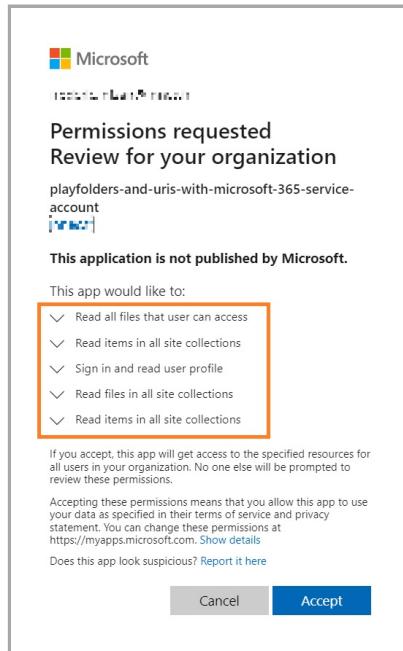
A login popup is displayed . Enter once your Microsoft 365 (<https://www.office.com/>) login credentials having a profile with rights granted for administration of Microsoft Application. This message is then displayed in a Powershell context.

You must log into an administrator account of your organisation and grant the necessary permissions.
A consent request will be sent within 30 seconds in your browser.

After thirty seconds, a login popup should be prompted (<https://login.microsoftonline.com/>) automatically in your default Web browser.

Enter again your Microsoft 365 login credentials.

A new popup message with the *Permission requested, review for your organisation* title is prompted in your Web browser.



Click on the `Accept` button. Then a message is displayed in your Web browser showing that the consent is successful: *Success of the consent request*.

You can view the data of the created application by typing the following command :

```
PS > $result
Name          Value
----          -----
clientId      xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
objectId      xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
spId          xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
name          playfolders-and-uris-with-microsoft-365-service-account
tenantId     xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
clientSecret xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

The result of the *Powershell* function is also stored in a JSON file: `result.json` .

Edit the file and store previously the values required for your application:

- the `clientId` value,
- the `tenantId` value,
- the `clientSecret` value.

Example to delete an Azure Active Directory application

```
PS > Remove-AADApplication -appname "playfolders-and-uris-with-microsoft-365-service-account"
```

A login popup is opened. Enter again your Microsoft 365 credentials. In case the values do not allow playfolders or uris with *Microsoft 365 services account* to work properly, check in Azure portal that the application has been created succesfully and the rights are properly granted. If not, wait for a while, the rights granting may take several hours.

6.6 Appendix: Teams

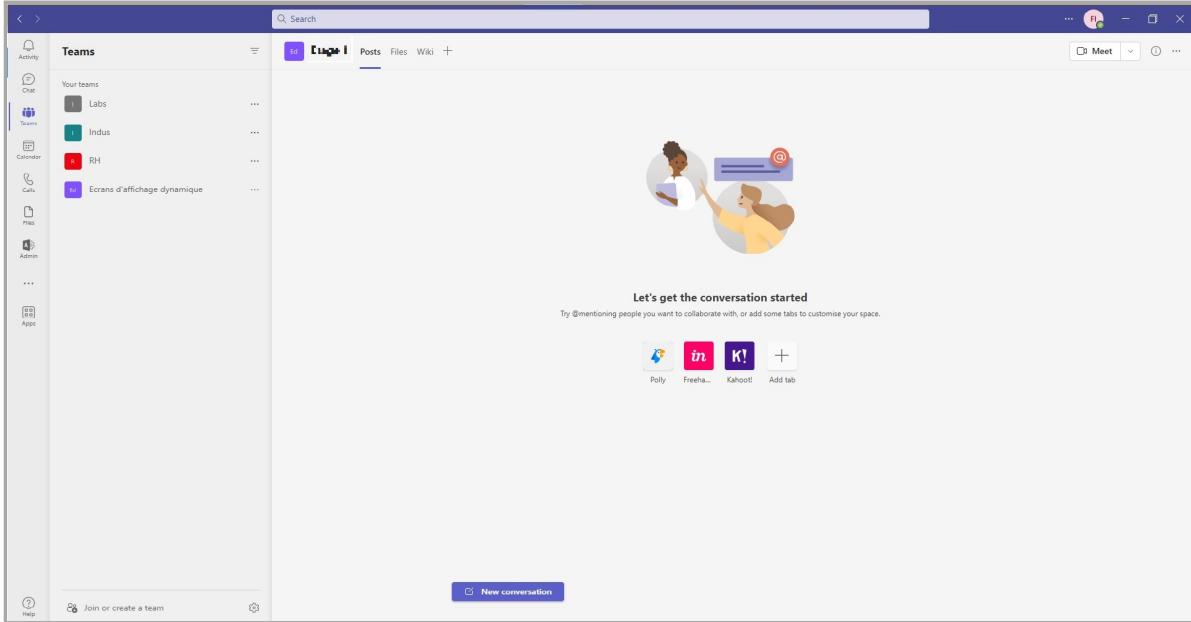
This is a way to organize your folders in Microsoft Teams to then work with playfolder configured with your Microsoft 365 services account.

We suggest you to create a new Team (e.g.: *Demo content*) having two channels (e.g.: *Floor1* and *Floor2*), whose content update access is granted for only some specific people.

Open your Microsoft Teams with an account having granted rights to administrate your Microsoft Teams.

At this step, the Team (e.g.: *Demo content*) is not yet created.

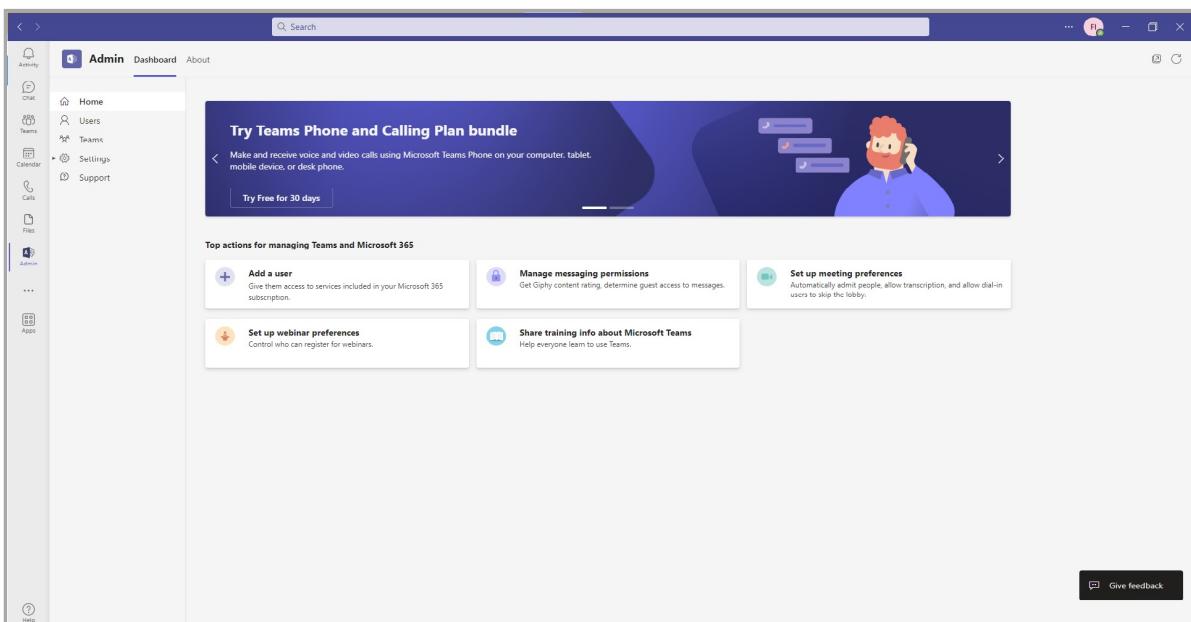
On the left banner, click on the Admin item.



On the Admin pane, several items appear:

- Home,
- Users,
- Teams,
- Settings,
- Support.

Click on the Teams items.



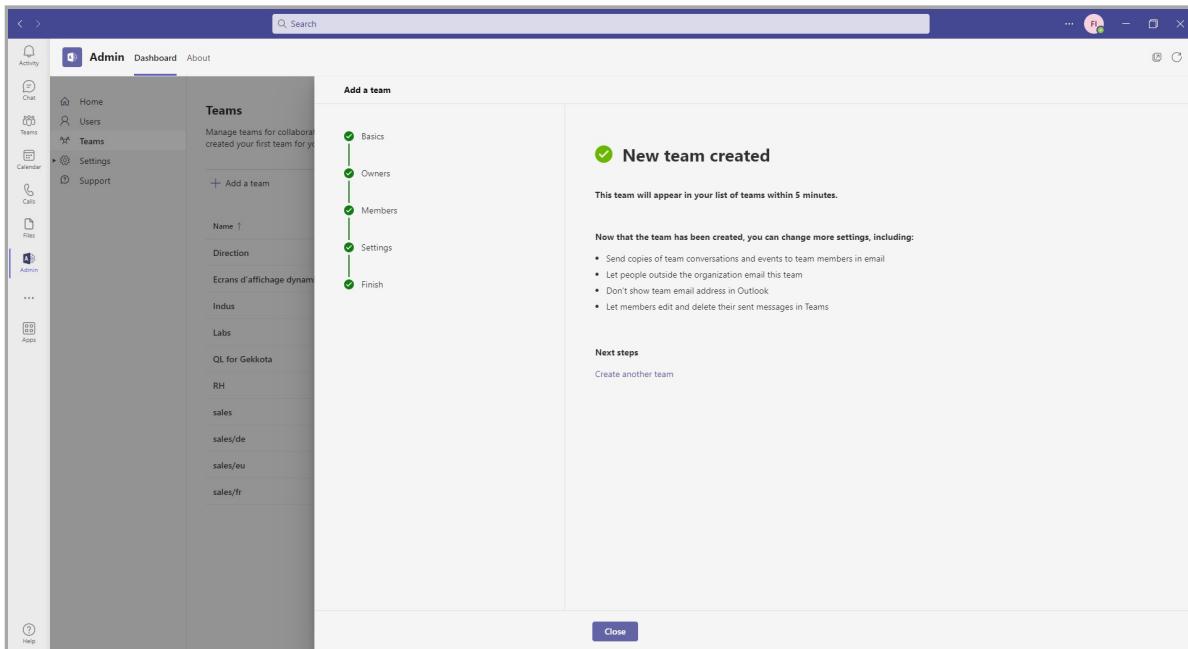
Click on the + Add a Team button.

The screenshot shows the Microsoft Teams Admin center. On the left, there's a navigation sidebar with options like Activity, Chat, Teams, Calendar, Calls, Files, Admin, and Support. The 'Teams' option is selected. The main area is titled 'Teams' and contains a sub-instruction: 'Manage teams for collaboration, chat, and meetings in Microsoft Teams, including members, owners, guest'. Below this, there's a section for 'Add a team' with fields for 'Name' and 'Email'. A list of existing teams is shown, including 'Direction', 'Ecrans d'affichage dynamique', 'Indus', 'Labs', 'QL for Gekkota', 'RH', 'sales', 'sales/de', 'sales/eu', and 'sales/fr'. At the bottom right of the main area is a 'Give feedback' button.

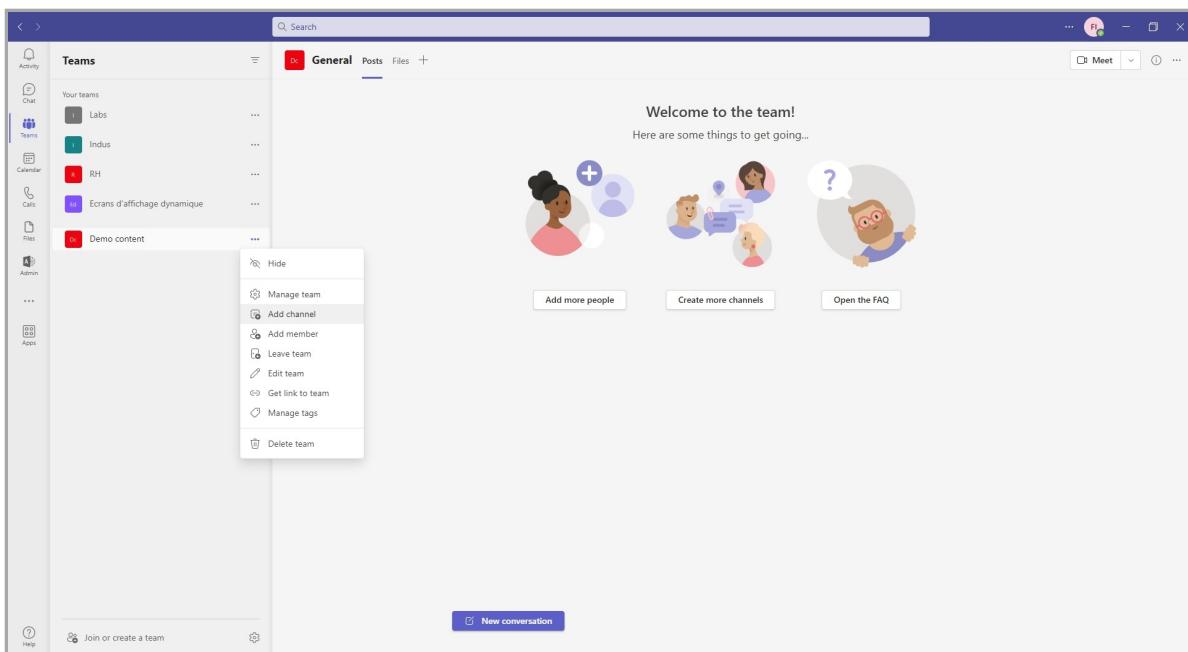
Enter a label for your Team (e.g.: *Demo content*). Click on the **Next** button and fill the appropriate new pane with the Team owner email address and the Team member email addresses.

This screenshot shows the 'Add a team' wizard. The left sidebar is identical to the previous screenshot. The main area is titled 'Set up the basics'. It includes a brief description: 'A team provides a place to chat and collaborate on files in Microsoft Teams, and an email address for reaching team members in Outlook. Learn more about Microsoft Teams.' There are four steps listed: 'Basics' (which is selected), 'Owners', 'Members', and 'Finish'. Under 'Basics', there's a 'Name of team' field containing 'Demo content' and a 'Describe this team' text area with the placeholder 'Contents for Demo M365'. At the bottom are 'Next' and 'Cancel' buttons.

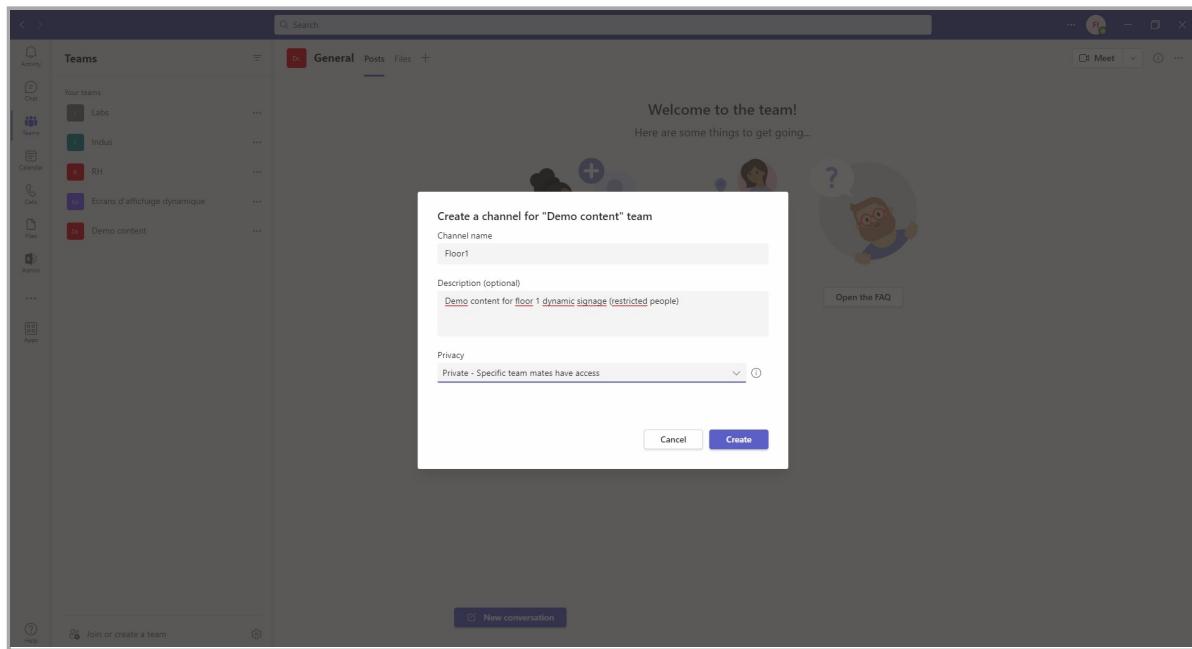
When the Team creation is completed is done, click on the Done button.



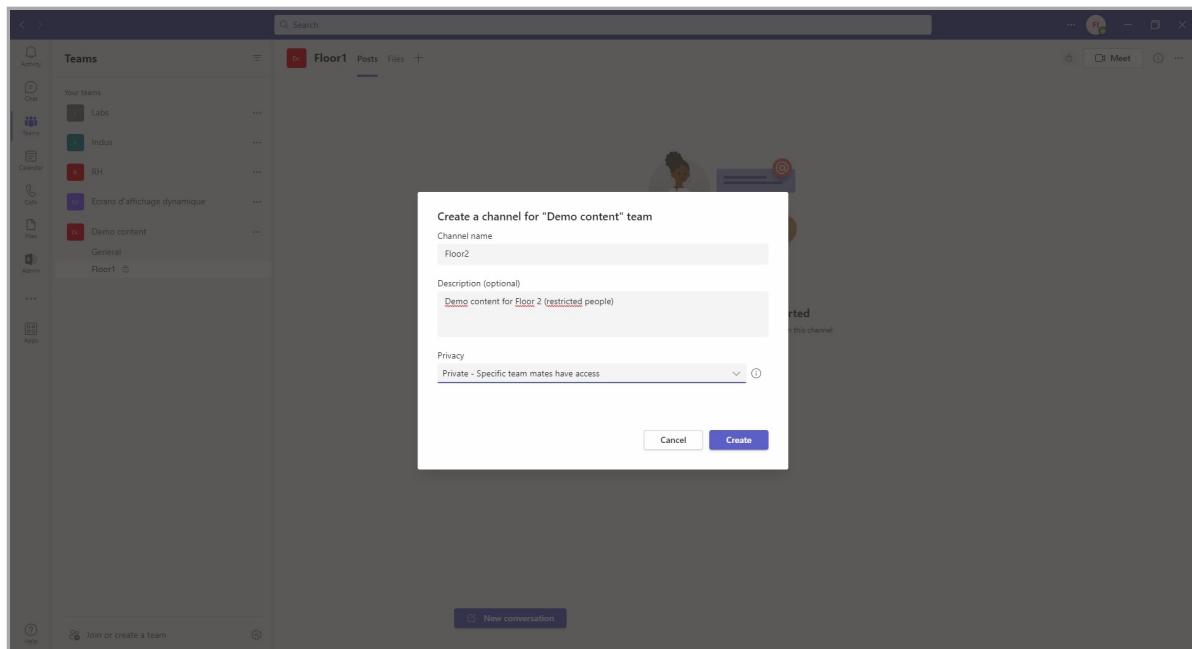
On the left banner, click on the Teams button and right click on the new Team that has just been created (e.g.: *Demo content*). Then select the Add channel item to add a *Floor1* channel.



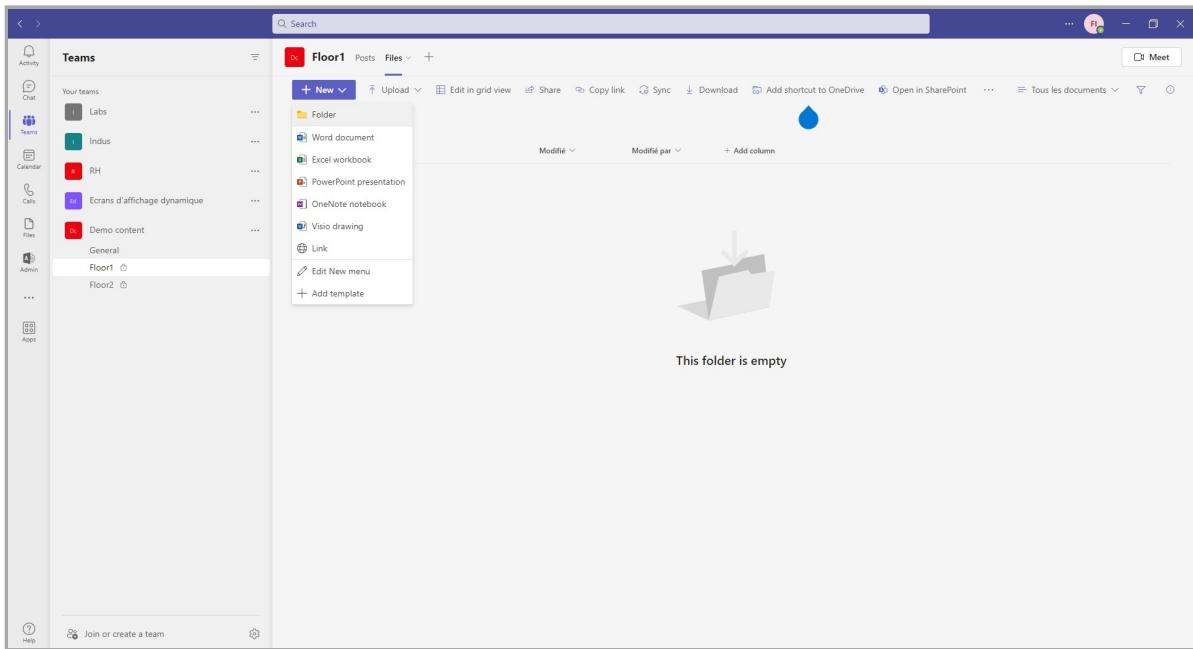
Enter a label for your channel (e.g.: *Floor1*) then select **Private - Specific Team mates have access**. Click on the **Create** button. When requested, enter the people accounts that must have access to this channel.



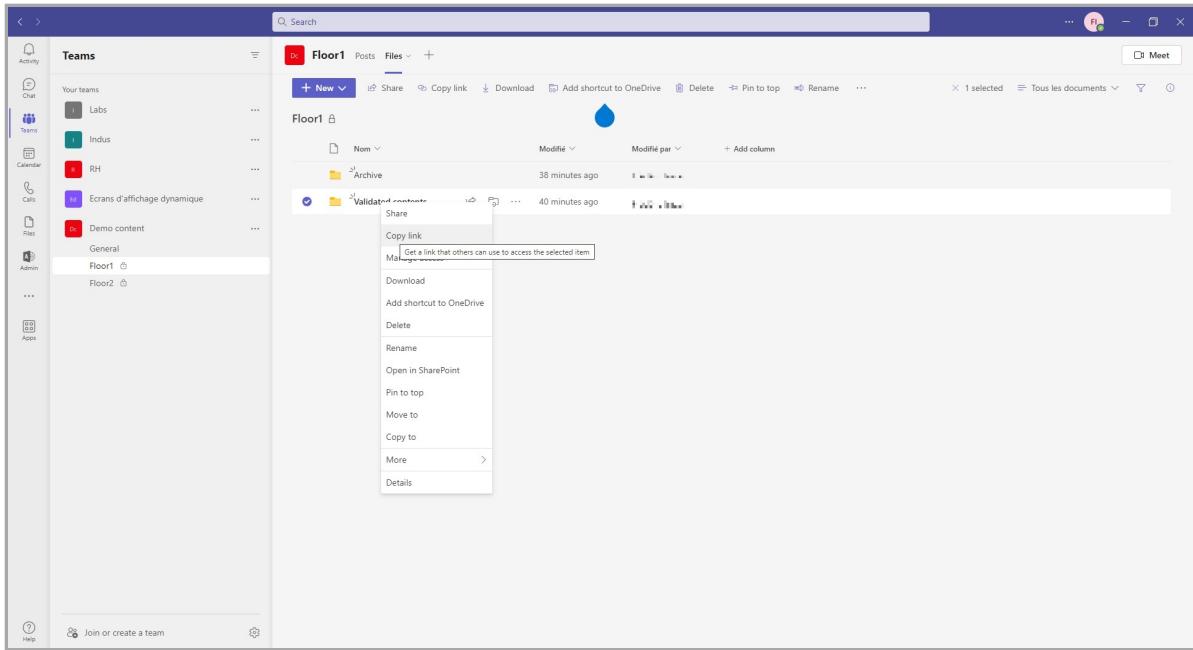
Do the same with the *Floor2* channel.



Click on the channel just created (e.g.: *Demo content > Floor1*). In the **Files** tab, click on the **+ New** button then in the list, click on the **Folder** item.



When the folder (e.g.: *Validated contents*) is created, right click on it and click on the **Copy link** item.



Paste the link to the shared folder in your playfolder configured with your Microsoft 365 services account[~] so that it can play the content of your shared folder.

Abound your shared folder (e.g.: *Validated contents*) with some medias.

Do the same with another playfolder configured with your Microsoft 365 services account[~] and with a Sharepoint[~] link to a similar directory (e.g. *Validated contents*) created in the second channel (e.g. *Floor2*).

6.7 Appendix: discovery & publication timings

Discovery strategy

When the device discovery is launched, in the default Screen Composer configuration, the devices have ten seconds to answer to Screen Composer they are alive in the network.

Publication strategy

When the publication is launched, in case the publication fails, a new publication attempt is done several seconds after for the involved device.

In case configuration with very much devices (e.g.: more than one hundred), the publication may fails randomly for some rare device whereas the devices are up on the network with a right MAC address value and with right credential values. In case the publication fails for a device, this publication status is displayed for the device: *The operation was not completed correctly because one of the recipients is not accessible*. A new publication on this device solves systematically the trouble.

Modify the publication strategy

To work around this unexpected publication trouble on some configuration with very much devices, it is possible to increase the number of publication attempts and delay the time between two publication attempts. Screen Composer can supports a `user.js` file allowing to configure some hidden user preferences. The default directory for this file is `C:\Users\<user1>\AppData\Roaming\INNES Screen Composer G4\user.js`.

 If not present, create manually the `user.js` file with your Windows explorer in this directory.

This is example of preferences values to publish on a configuration with very much `SAP10e` device.

hidden user preference	Information	Default value	Suggested value for configuration with very much device
<code>innes.network.publishing.retrycountmax</code>	Number of connection attempts in case connection failure to a device	2	
<code>innes.network.publishing.retry_delay</code>	Delay between two connection attempts in case connection failure (in second)	20	
<code>innes.network.publishing.networkfailuredelay</code>	Timeout duration allowing to the App to consider that the connection to a device has failed (in second)	45	

Paste these three lines in the `user.js` file, modify the value according to your need and save it. Close `Screen Composer` and launch it again.

```
pref("innes.network.publishing.retry_count_max", 2);
pref("innes.network.publishing.retry_delay", 20);
pref("innes.network.publishing.network_failure_delay", 45);
```

Default configuration

The values specified below are those of the default Screen Composer configuration.

hidden user preference	Information	Default value	Suggested value for configuration with very much device
<code>innes.network.publishing.retrycountmax</code>	Number of connection attempts in case connection failure to a device	1	
<code>innes.network.publishing.retry_delay</code>	Delay between two connection attempts in case connection failure (in second)	2	
<code>innes.network.publishing.networkfailuredelay</code>	Timeout duration allowing to the App to consider that the connection to a device has failed (in second)	10	

If you want to return to the default publication parameter value, paste these three lines in the `user.js` file, modify the value according to your need and save it. Close `Screen Composer` and launch it again.

```
pref("innes.network.publishing.retry_count_max", 1);
pref("innes.network.publishing.retry_delay", 2);
pref("innes.network.publishing.network_failure_delay", 10);
```