INNES

Application note

Player synchronization

Version 006C

March 2017

Table des matières

1 Aim of document

The aim of this document is to explain how to configure different players to be synchronized each other. This feature is used generally to synchronize player for a screen wall

- Playing the same video on all the screens
- different spatial part of only one video on the screen wall

2 Pre-requisite

2.1 Minimal versions

- Screen Composer G3 3.11.12
- PlugnCast server 3.10.24 (or above)
- Script xpfSyncManager-V1.10.13.js (or above)

2.2 Player family: SM3

Only the players SMA300 support the player synchronization with a frame cue accuracy.

These version of Gekkota OS SMA300 are required to support properly this functionality:

- From V3.12.31 to V3.12.37 only

2.3 Local network

To be synchronized, the players need to be installed on the same local network.

2.3.1 Local NTP server

In order to avoid any unexpected issue on clock estimation, you must use a local NTP server (instead of Web NTP server).

Note: the synchronization feature is not accurate and stable enough when a Web NTP server is used.

2.3.2 UDP multicast

The local network has to support UDP multicast because the communication between the players for the synchronization feature uses the UDP protocol.

2.3.3 Overload network

An overloaded network can lead to unexpected bad effects on clock estimation error.

3 Player configuration

3.1 WebUI: NTP configuration

The NTP server needs to be activated and properly configured in player WebUI (trial number, delay between trials).

When the NTP server is activated, the player clock precision error at boot-up is around 200ms*.

** In case network overload, the clock estimation error can be larger.*

- In case distant NTP server, it is advised to configure
  - 3 trials with 30 seconds between each trial
  - 5 trials with 60 seconds between each trial (worst network condition)
- Innes recommends to use a local NTP server (time & date updated most of time by GPS with very good accuracy)

- It permits to reduce very much all the side effects linked to the internet network poor quality (problems linked to latency delay for uploading/downloading requests with the NTP server)
- For more information, contact support@innes.pro

Note: a MS Windows PC can behave as NTP server when NTP is activated. Refer to "Activate NTP server on MS Windows" application note for more information

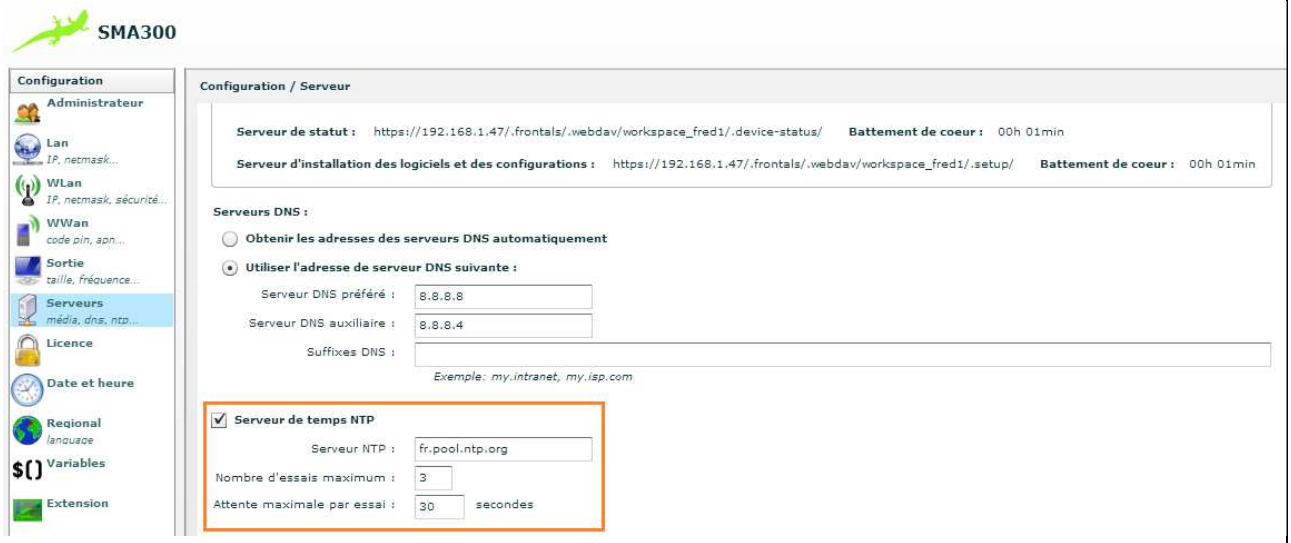
In this example, NTP server is configured with URL of a distant NTP server

- NTP Server: "fr.pool.ntp.org"

In case a local NTP server is available, enter the URL of your local NTP server according to in the WebUI.

- NTP trial number: 3
- Delay between retries: 30 seconds

Note: NTP can be activated also with auto-configuration file.
Refer to Gekkota application note auto-configuration V1.10.19 (or above) for more information



3.2 Slight clock synchronisation with NTP server

The player is getting NTP clock (time&date) at player boot-up. The time must be constantly adjusted afterwards to ensure that the content is played synchronized.

To activate clock synchronization with NTP:

- Inject auto-configuration V1.10.19 (or above) with the function **setClockSyncEnableNtp** activated (see next paragraph)
- After a reboot, the player is starting quick & coarse clock correction process, and then, continue with low & accurate clock correction process. This clock correction convergence can be few long. The time duration to compensate clock drift (and reach clock drift error lower than 50 milliseconds) can be from 1 hour to 3 hours (this duration is depending on network environment).

3.2.1 Auto-configuration file: setClockSyncEnableNtp

- Download auto-configuration script V1.10.19 (or above) from INNES support site
- Open the configuration script and activate the function:
  - **setClockSyncEnableNtp**()
    - *Refer to Gekkota application note auto-configuration for more information*

3.2.1.1 By USB injection

- Save the file into 000000000000.js on USB stick

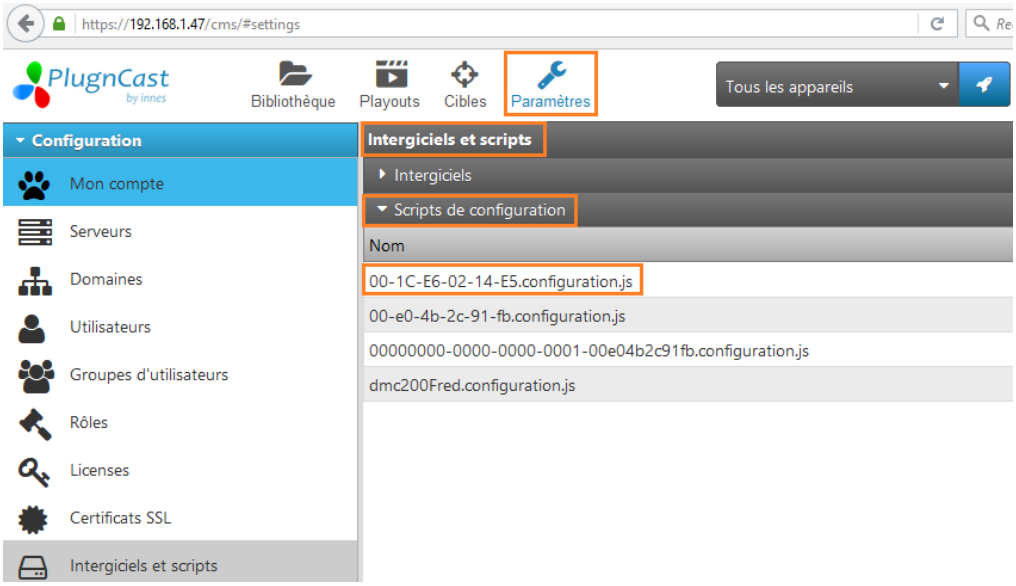
- Inject it in all SMA300 you want to synchronize

Refer to Gekkota application note auto-configuration V1.10.19 (or above) for more information

3.2.1.2 By http download (PlugnCast G3)

- Prerequisite: yours players are configured in “PlugnCast” mode
- Register your players on a PlugnCast G3 server
- Save the file into configuration.js, and import it in settings/configuration scripts

Refer to Gekkota application note auto-configuration V1.10.19 (or above) for more information



- PlugnCast G3 > target: activate all your SMA300
- PlugnCast G3 > target > system: click on button “deploy configuration script” and wait for a while in order the player has time enough to take the script

4 Playout configuration and media behaviour

4.1 Playout

The player synchronization works properly when the players are playing:

- Only one video media
  - With behaviour: “infinitely”
  - With same media features (codec, frame number, frame rate, GOP, duration…)
- Only one URI playfile (video media) (variable supported)
  - With behaviour: “infinitely”
  - With same media features (codec, frame number, frame rate, GOP, duration…)

4.2 Audio of AV stream not played

In case the AV stream contains audio track, the audio track is not played to avoid any unexpected audio-video synchronization issue.

In order to not allow audio track playing of audio-video stream:

- **Screen Composer**:
  - Activate an audio zone, but don’t drop an audio media in audio channel
- **PlugnCast G3**:
  - Create a video channel (no audio channel)

4.3 Delay between reboot time and time when the media need to start

Given that player can takes from 1 hour to 3 hours to reach clock accuracy with an clock drift error lower than 50ms*, it is advised to keep a consequent delay between the reboot time and the time when the media begins to be played perfectly synchronized.

Recommended use case

- make reboot all the players at 00h00 (and play nothing until the time when the media to start perfectly synchronized)
- start video at 8.30 AM

** In case network overload, the clock drift correction can face issue to work properly. So it is advised to keep extra time between reboot and start of the media*

4.3.1 Player not completely synchronized after reboot

In case a reboot occurs while the video is currently played, the clock drift correction is forgotten and a new clock drift correction process is starting again. So, in such cases, a slight video delay can be noticed. To correct it, publish again to force the player to restart their playout synchronized.

4.4 Sequence of 6 hours max (to reset additional drift in video chain)

An additional disparity of around 150ms could be noticed randomly in SMA300 player video chain after 6 hours of synchronized video making that one or several players could be slightly not synchronized. To reset this disparity, it is advised to restart the sequence every 5 to 6 hours maximum.

4.5 Add xpfSyncManager-V1.10.13.js to a playout

To complete the synchronization process, you need to add a script to the playout.

This script forces players to restart their video (or URI playfile video) around 30 seconds after a reboot or a publication.

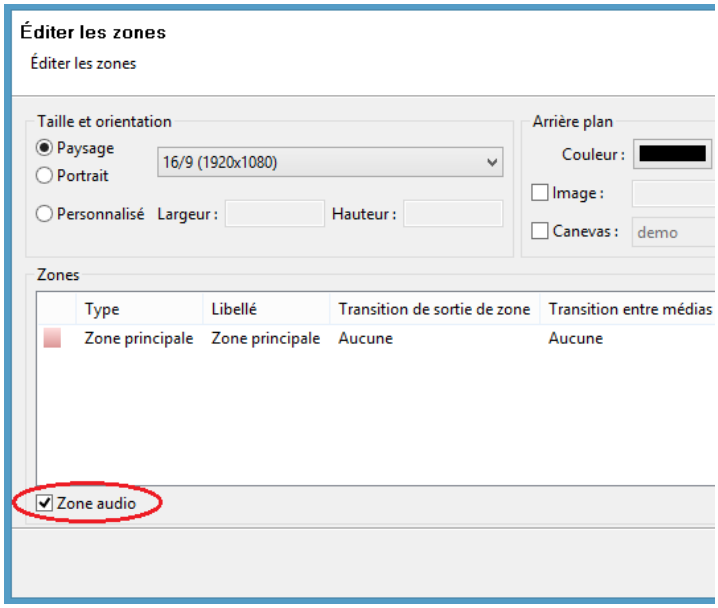
- When a player reboots, it forces all the others to restart to play at the beginning of the content.
- When a publication happens, the script is doing the same also.

That permits for several player to start to play a video media perfectly at the same clock cue.

Note: The script is using UDP multicast messages. The UDP multicast address can me modified inside the script if required.

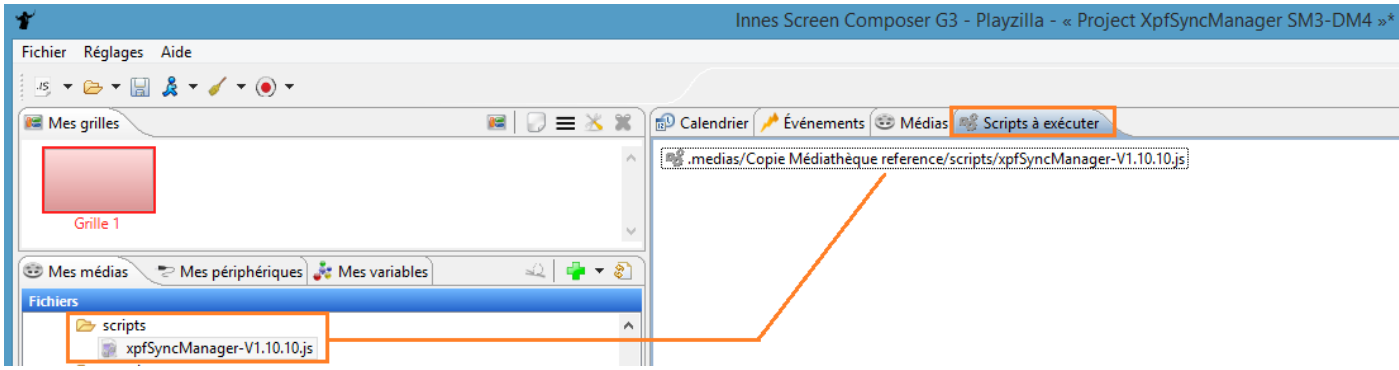
4.5.1 Screen Composer

- Create a new project "**playoutSync**" with Screen Composer.
  - Add your players SMA300 to your project and register them
  - Activate "**publishing**"
- Edit the zone of the grid and activate zone audio



- Import a video media (ex video_test1.mp4) (or URI playfile* of video media) in the Screen Composer library and
  - Drop it in the calendar view
    - Sequence1 from 8h00 AM to 2h00 PM (6 hours max)
    - Sequence2 from 2h00 PM to 8h00 PM (6 hours max)
  - Apply a recurrence (planning / every days) for the 2 sequences
  - Apply a media behaviour "**played infinitely**" for the video media for the 2 sequences
- Program a task player reboot every days at 0h00,
  - Ensure that no media is inserted in calendar from 0h00 to 8h00 (letting time for the player to reach accurate clock)
- Import script xpfSyncManager-V1.10.13.js in the Screen Composer library and drop it in tab "**script to execute**"

  In order to use URI playfile of video media with variable, refer to appendix

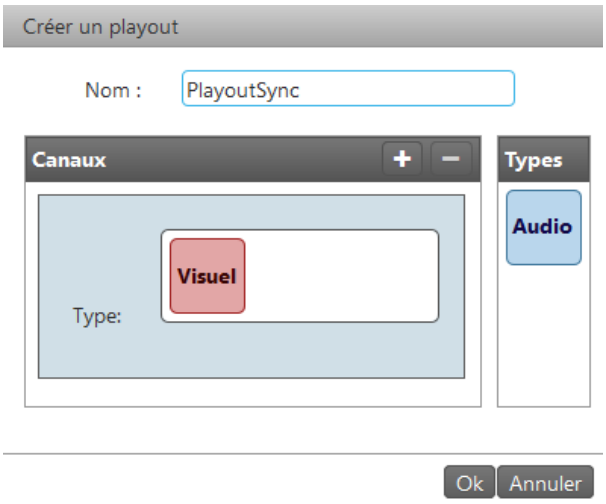


- Publish to all your players SMA300

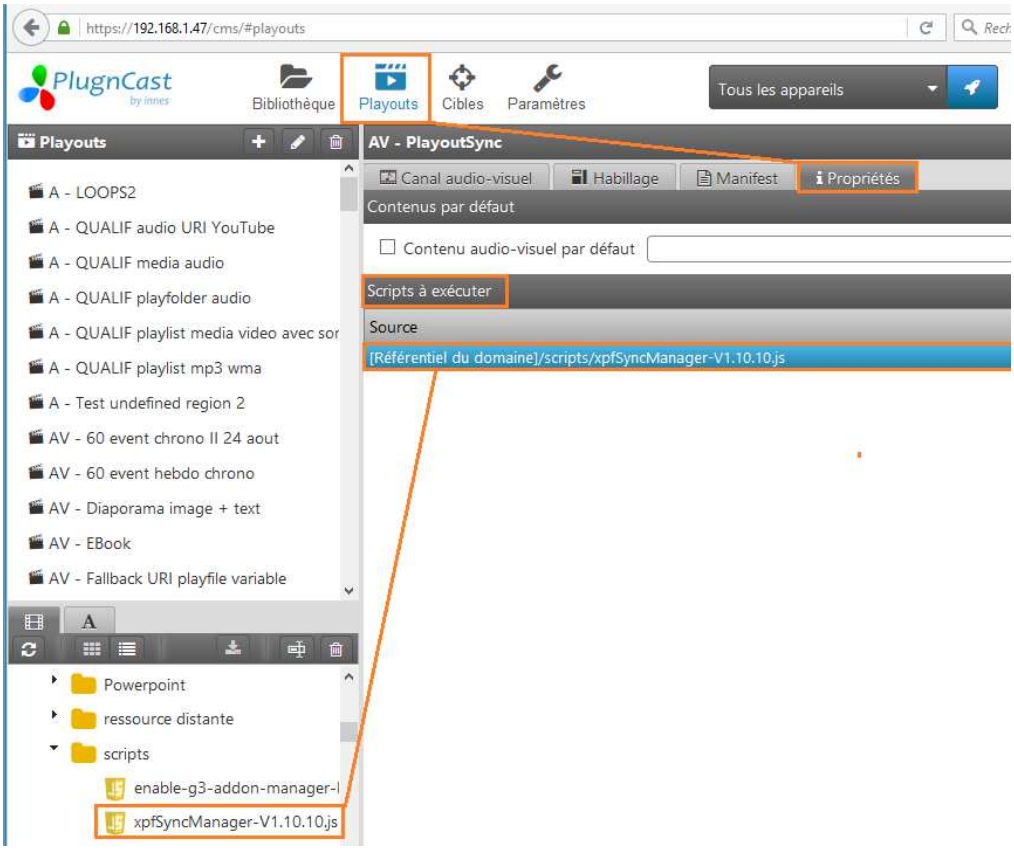
  Note at each publication, the players will restart their video once or more after a delay of few seconds

4.5.2 PlugnCast G3

- Create a new playout visual "**playoutSync**" in PlugnCast G3.



  - Register all your players SMA300
- Import a video (ex video_test1.mp4) in the PlugnCast G3 domain repository and
  - Drop it in the calendar view
    - Sequence1 from 8h00 AM to 2h00 PM (6 hours max)
    - Sequence2 from 2h00 PM to 8h00 PM (6 hours max)
  - Apply a recurrence (planning / every days) for the 2 sequences
  - Apply a media behaviour "**played infinitely**" for the video media
- Program a task player reboot every days at 0h00
  - Ensure that no media is inserted in calendar from 0h00 to 7h00
- Import script xpfSyncManager-V1.10.13.js in the Screen Composer library and drop it in tab "**script to execute**"



- Publish the all the SMA300 players

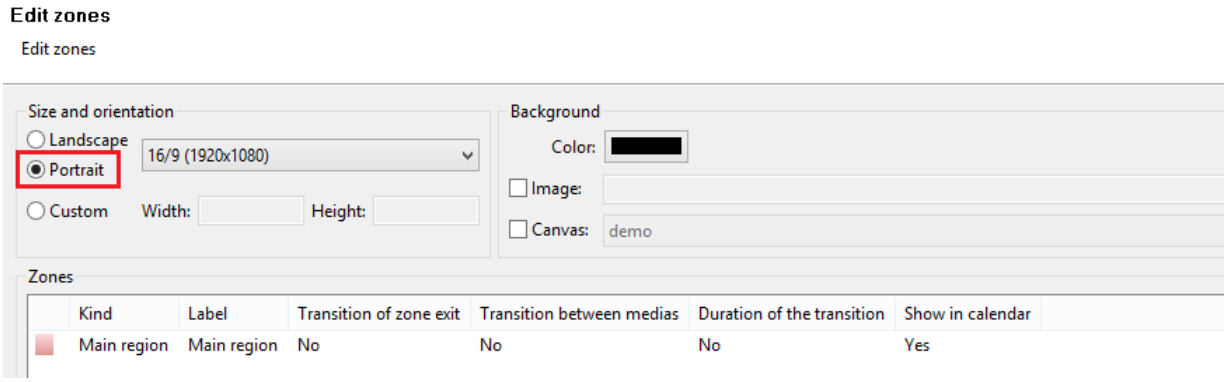
- At each publication, the players will restart their video once (or more time) after a delay of few seconds

5 Appendix

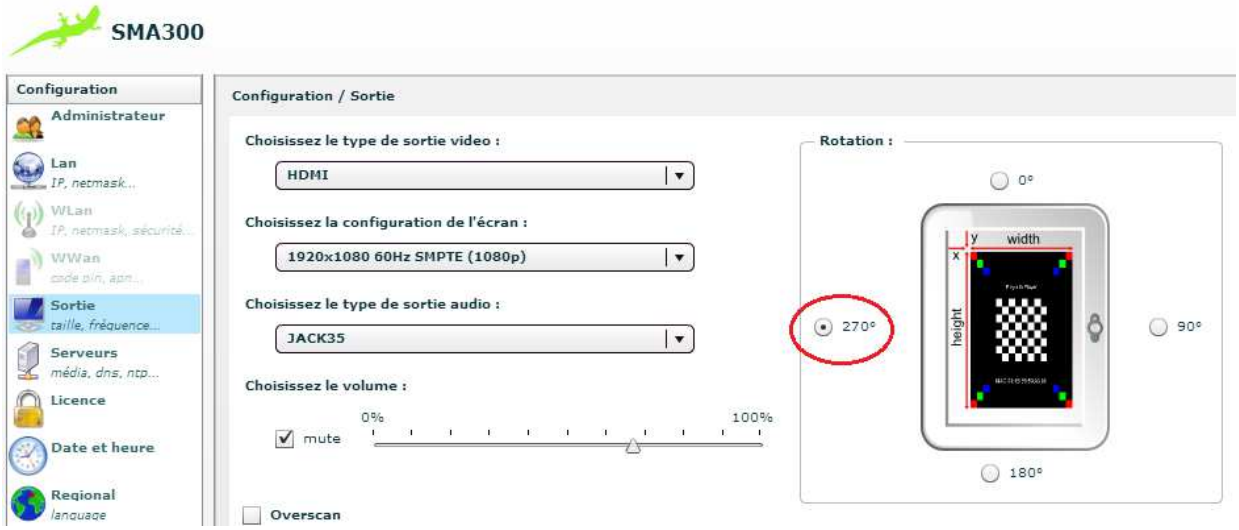
5.1 Screen wall portrait

In case the screen wall is composed of TV screens in portrait and the video medias have portrait aspect ratio,

- In CMS (Screen Composer G3 or PlugnCast G3), apply portrait aspect ratio for the grid



- With the player WebUI, apply a rotation of 90° (or 270°)



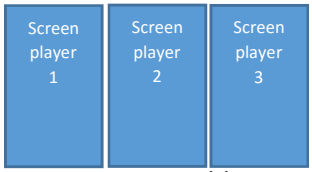
5.2 URI playfile for Screen Wall

Screen wall can be managed with only one project, and only one publication by using URI playfile (of media video).

An URI playfile video is a short cut to a video media of the library containing a rule (variable) to play the right media file.

5.3 How to with Screen Composer G3

Use case: tryptic 3 videos

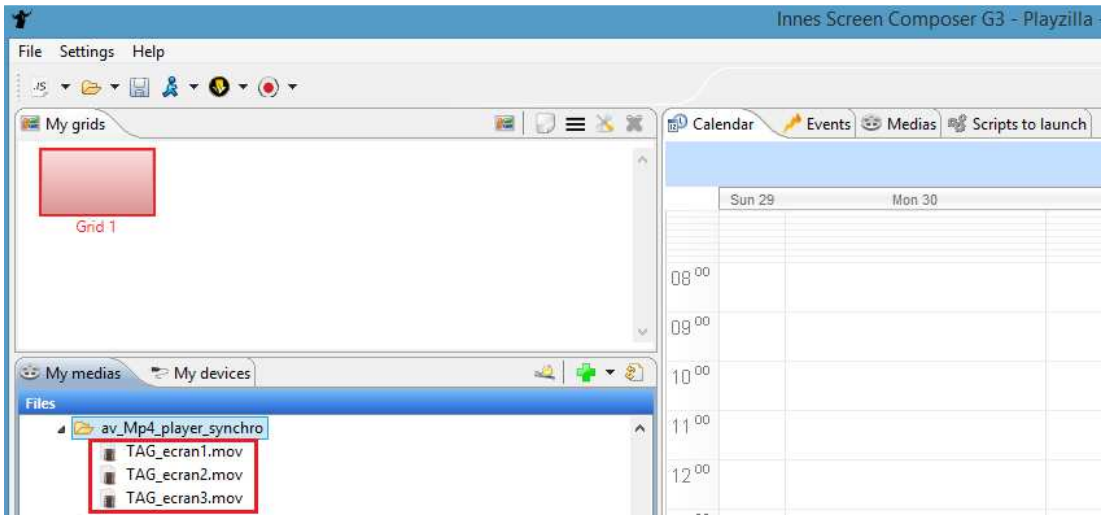


With Screen Composer, it is possible to manage a screen wall with only one playout, thanks to variable management.

5.3.1 Variable management (name and value)

Ex: in a directory, I own 3 video media whose name has suffix (ex: TAG_ecran1.mov)

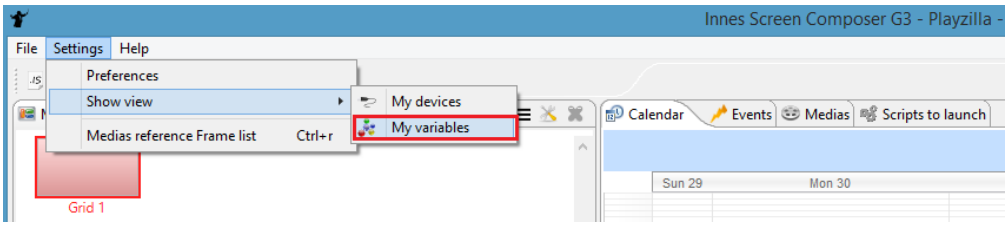
- _ecran1
- _ecran2
- _ecran3



Create a variable “numeroecran” which can have 3 values:

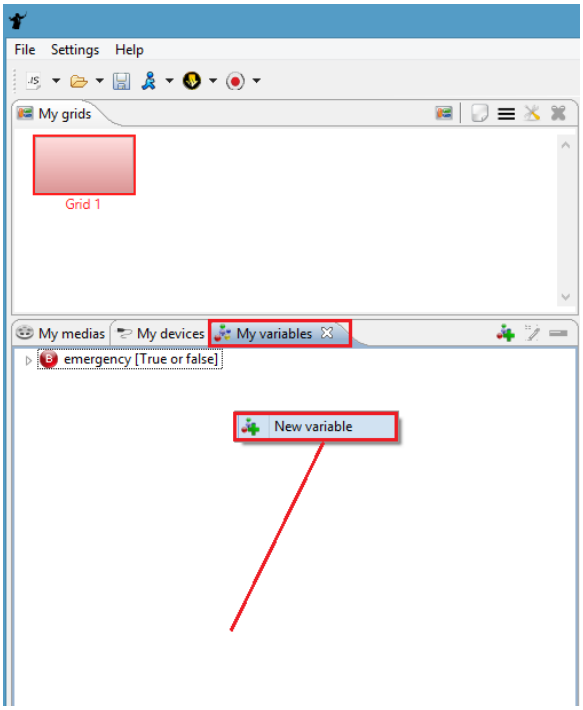
- _ecran1
- _ecran2
- _ecran3

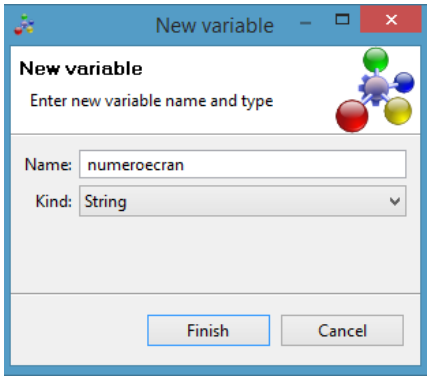
Display the variable view (not displayed by default): settings > Show view > My variables



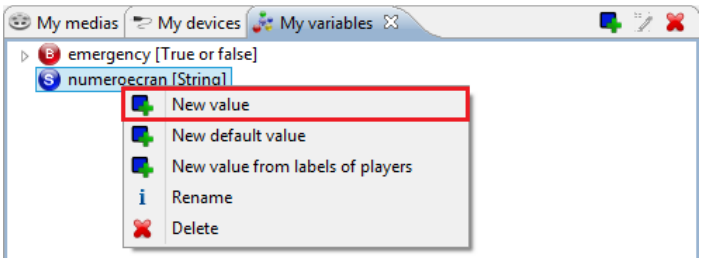
Select My Variables tab, right click on background, and select New Variable

Select a “name” and “kind” for the variable (ex: name = “numeroecran” and kind: “String”)

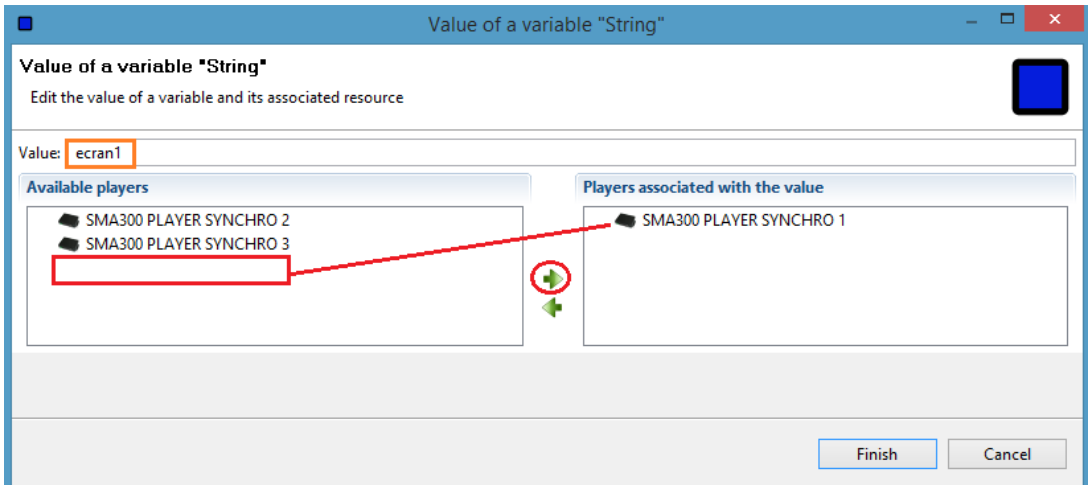




Add new value for the variable: select “numeroecran” and right click

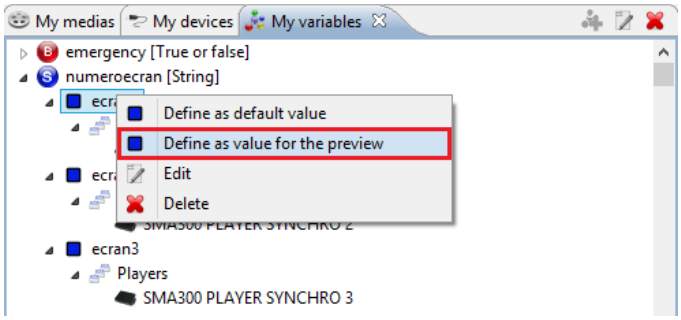


Enter a value (ex: ecran1) and affect this value to a player by a drag'n drop of the player 1 from the left to the right.



Apply the same for all the others players.

Set one of the value as value for the preview (to help to configure the URI playfile afterwards)



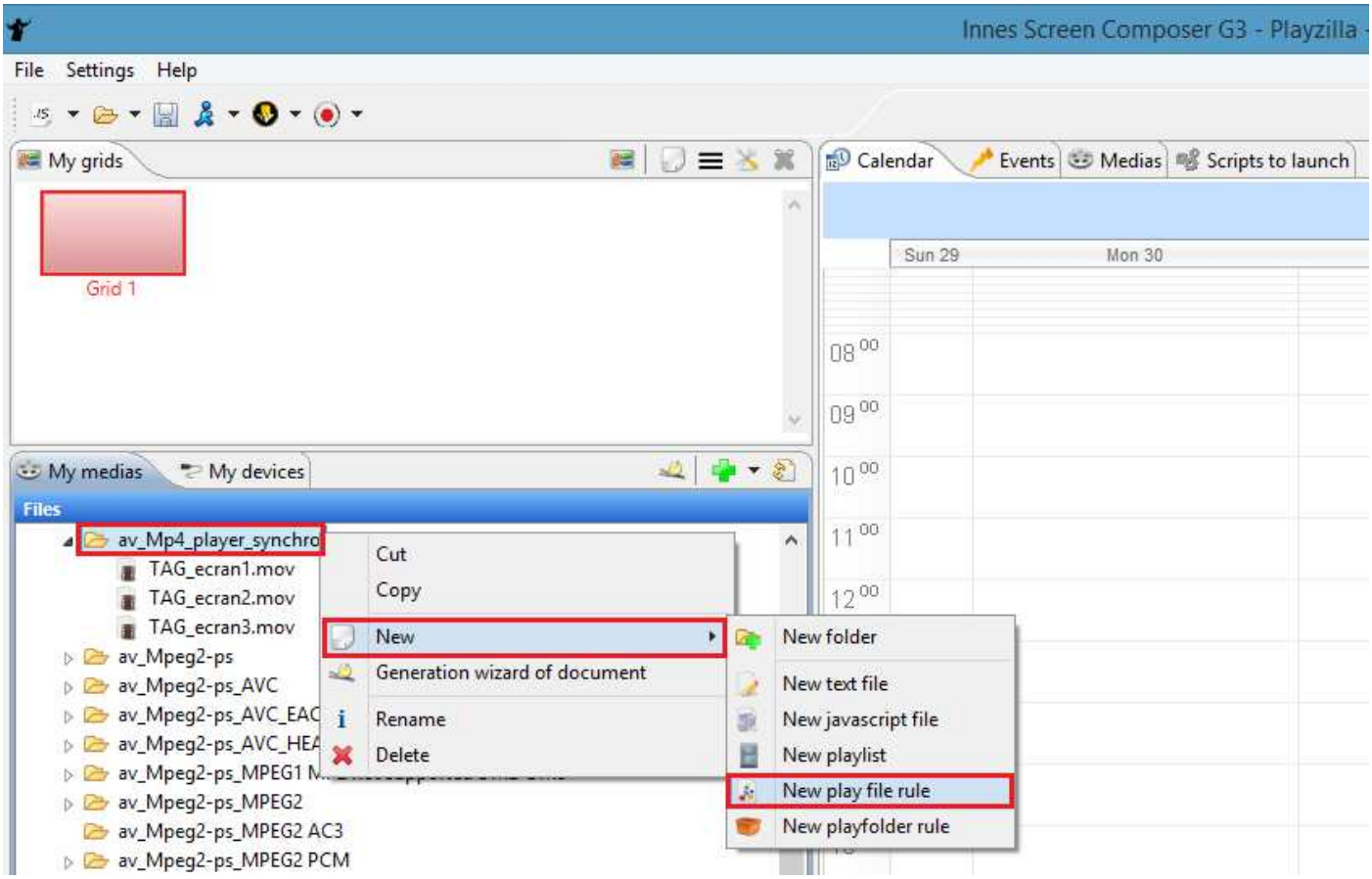
Recap for variable name (ex: numeroecran)

Variable value	Player name	value for the preview
- **ecran1**	Player1 (ex: SMA300 PLAYER SYNCHRO1)	X
- **ecran2**	Player2 (ex: SMA300 PLAYER SYNCHRO2)	
- **ecran3**	Player3 (ex: SMA300 PLAYER SYNCHRO3)	
- **...**	- ...	

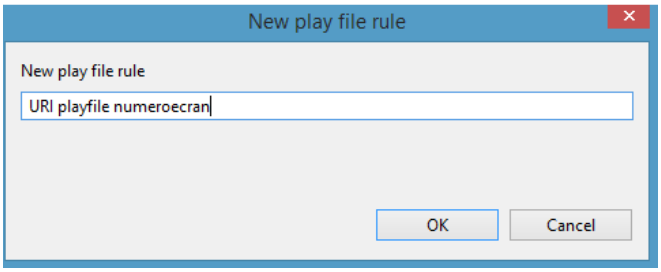
5.3.2 URI playfile (video media) creation (containing a variable)

Create an URI playfile to play one of the video media depending on the variable of the player

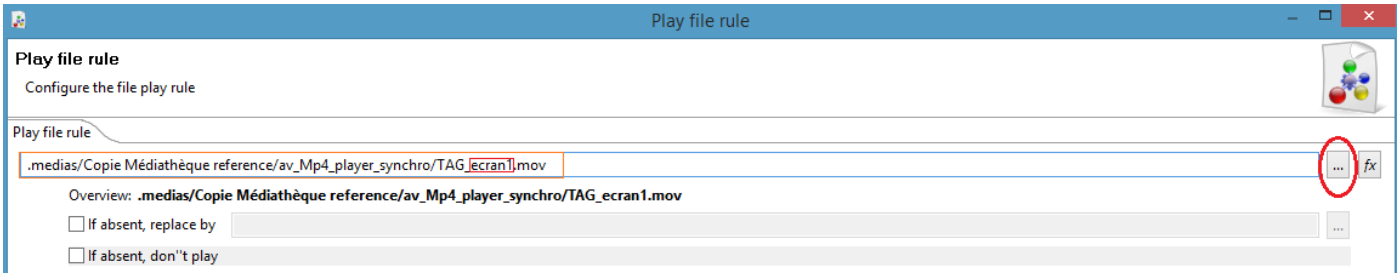
In My medias tab, select a directory to store the URI playfile, right click, select New > New play file rule



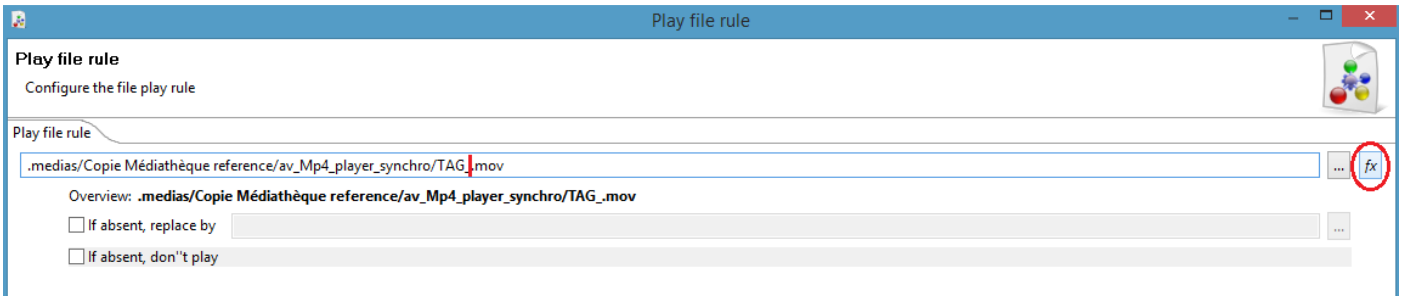
Give a name for your URI playfile (this media will be inserted directly in the calendar)

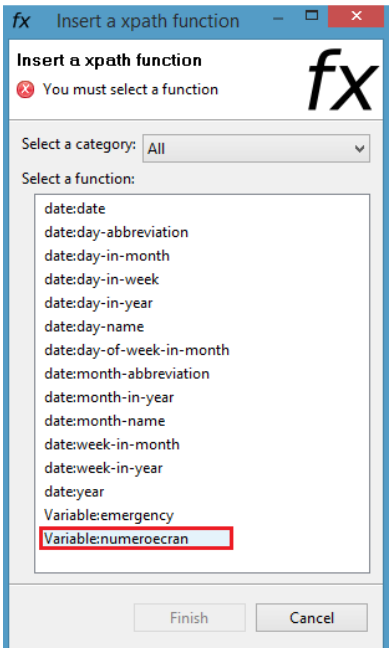


Select your media with "…" button. Remove "ecran1" part with "DEL" key.

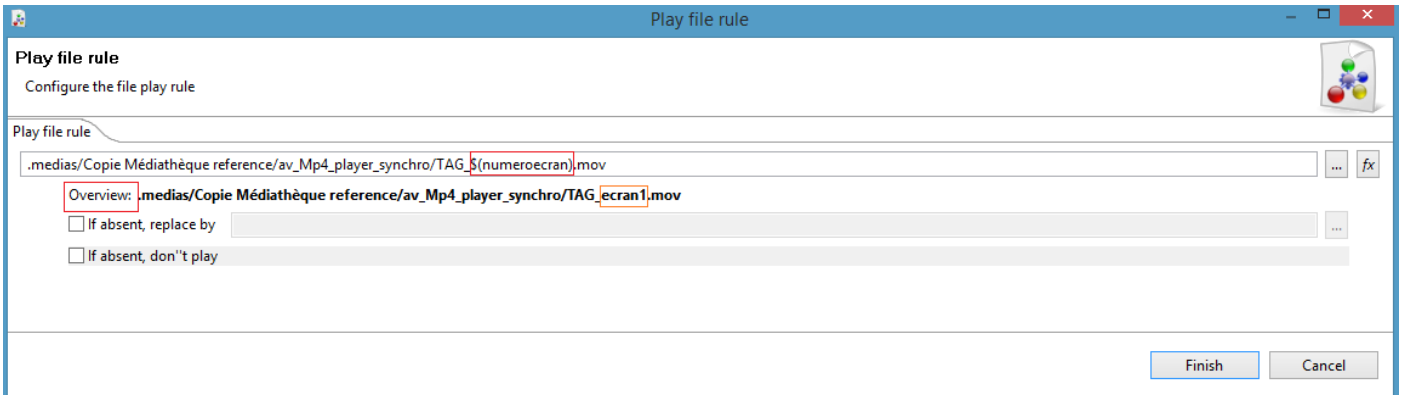


And insert in the URI the variable (ex: "numeroecran") with "fx" button





It should look like this:



Once publish into all player, each player should play the right media.

5.4 How to with PlugnCast G3

Use case: tryptic 3 videos

With PlugnCast G3, it is possible to manage a screen wall with only one playout, thanks to variable management.

5.4.1 Variable management (name and value)

Ex: in a directory, I own several video media of same name and whose name has suffix below (ex: TAG_ecran1.mov)

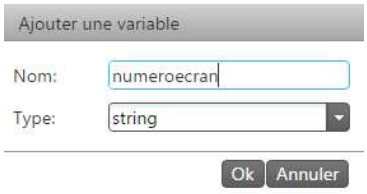
- “ecran1”
- “ecran2”
- “ecran3”

Create a variable “numeroecran” which can have 3 values:

- “ecran1”
- “ecran2”
- “ecran3”

Create a new variable: Librairie > Variables > Personnalized > Add a variable

Select a name and a type for the variable (ex: name = “numeroecran” and type: “String”)



Add new value for the variable: select “numeroecran” and on the right panel, click on add a button **+ “**add value”

Enter a value (ex: ecran1)

Create as much value as there is player on the screen wall. It should look like this:

Default first variable created is set as

- Default value
- Default for preview

Recap for variable name (ex: numeroecran)

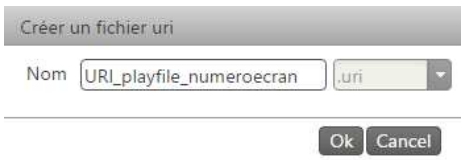
Variable value	Player name	value for the preview	Default
"ecran1"	Player 1 (ex: SMA300 PLAYER SYNCHRO1)	X	X
"ecran2"	Player 2 (ex: SMA300 PLAYER SYNCHRO2)		
"ecran3"	Player 3 (ex: SMA300 PLAYER SYNCHRO3)		
"…"	Player n…		

5.4.2 URI playfile (video media) creation (containing a variable)

Create an URI playfile to play one of the video media depending on the variable of the player

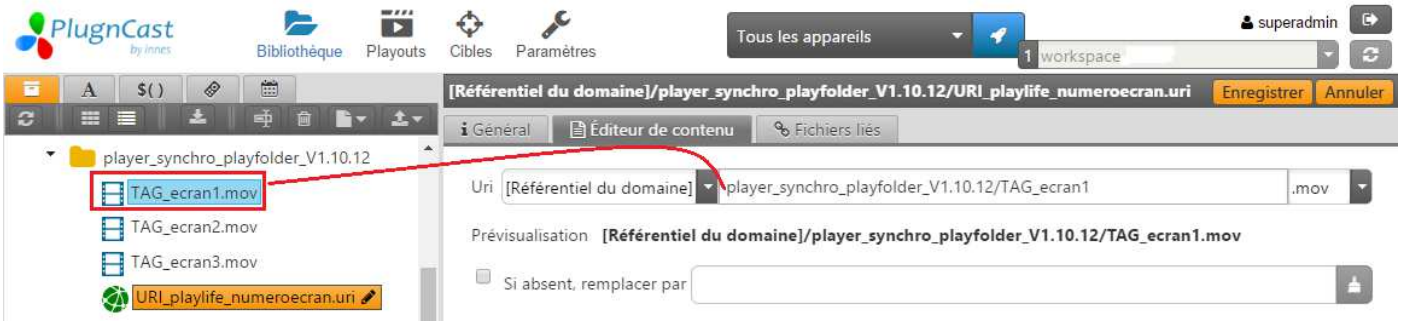
In Librairie > Files > Domain repository, select a directory to store the URI playfile. Then Add > New URI

Give a name for your URI playfile (this media will be inserted directly in the calendar)



Drag'n drop your video media.

Remove "ecran1" key word with "DEL" key from URI.



Select variable tab, drag'n drop your variable (ex: numeroecran) as suffix of URI

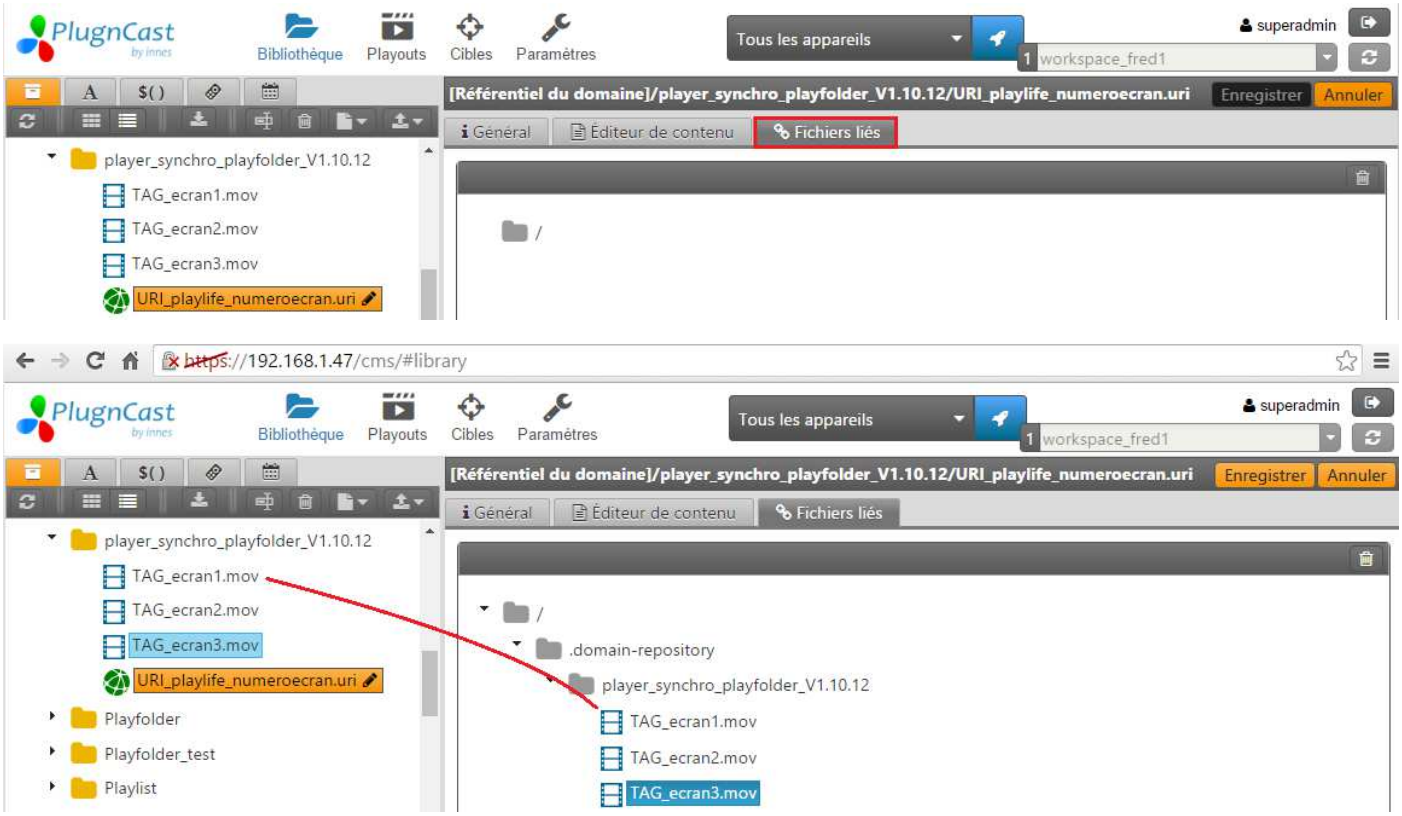
A preview is available in the screen to check the final rendering. Preview value is depending on preview value selected in variable interface

5.4.3 Linked media (very important)

Plugncast don't raise any publication issue when linked media are empty but for playfile using variable, linked media need to be filled with target medias.

Select URI, and on the right panel, click on "Linked media" then "modify".

Then drag 'n drop all the video required



Then save.

Note: with PlugnCast 3.10.32 (or above), linked medias is resolved automatically by PlugnCast G3. It is not required to add the video medias in linked medias tab

5.4.4 Affect a variable to a player

The affectation of a variable value is done at the publication.

Select target interface (1)

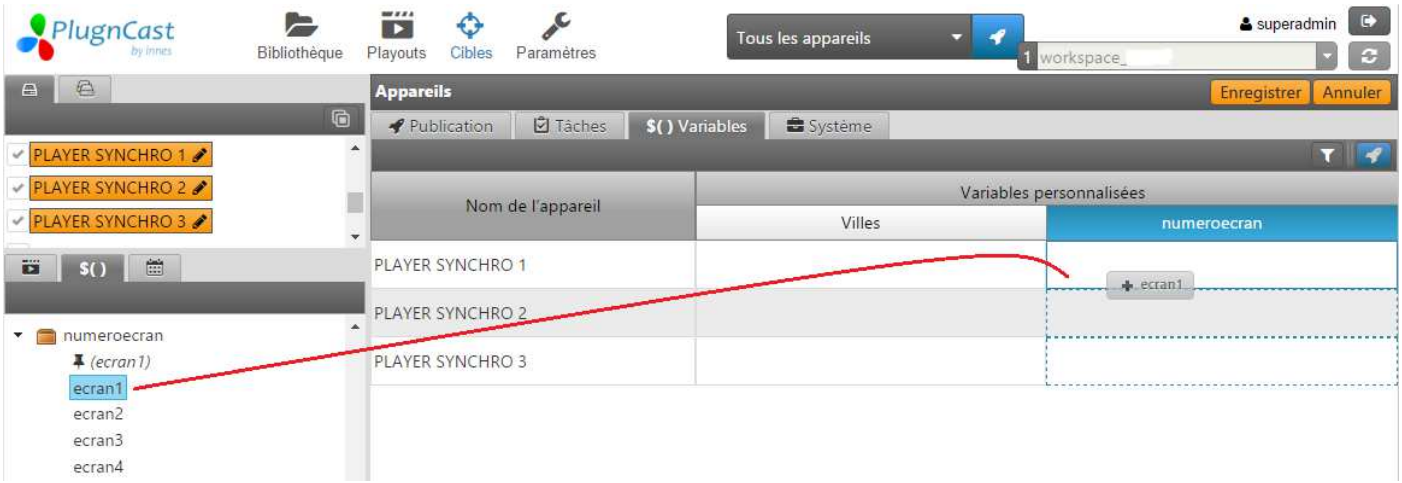
Check that the player of the screen wall are selected (2 and 3)

Open target > variable tab (4 and 5)

Display target > Variables tab (6) and check that column of your variable (ex: “numeroecran”) present (8)

If not present, click on button select the variables (7) and select your variable.

Then affect a variable to a player by a single drag’n drop



Do the same for all the player of the screen wall

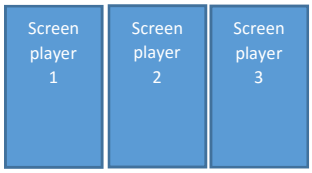
Recap for variable name (ex: numeroecran)

Variable value	Player name
“ecran1”	Player 1 (ex: PLAYER SYNCHRO 1)
“ecran2”	Player 2 (ex: PLAYER SYNCHRO 2)
“ecran3”	Player 3 (ex: PLAYER SYNCHRO 3)
“…”	Player n…

Click on save. Once playout + script + variable published into all player, each player should play the right media.

5.5 How to with PlugnCast G2

Use case: tryptic 3 videos

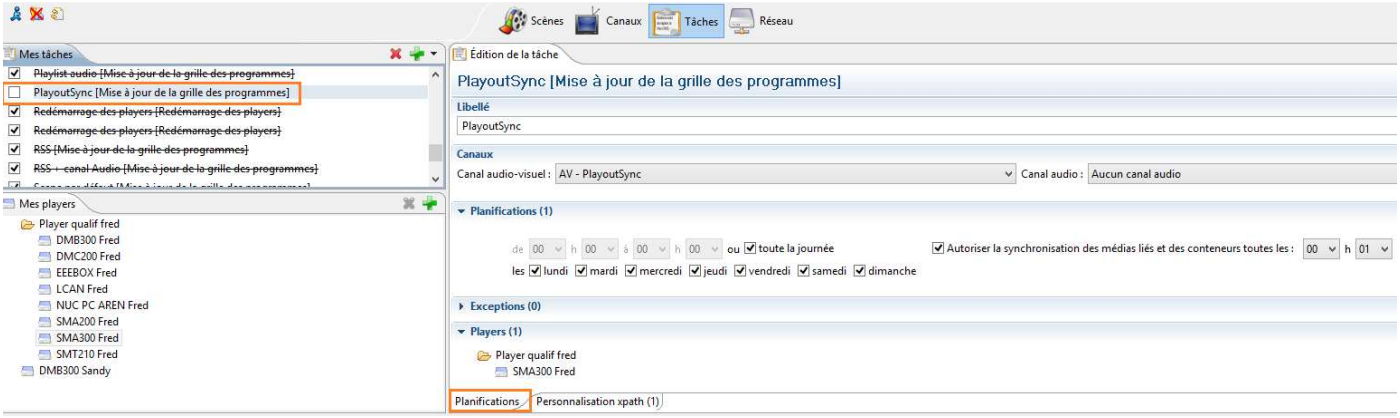


With PlugnCast G2, it is NOT possible to manage a screen wall with only one playout and all players synchronized because opt-outs are not supported by script-V1.10.13. The only way is to create 3 playouts.

- Import script xpfSyncManager-V1.10.13.js in the PlugnCast G2 library
  For example in : Script/xpfSyncManager-V1.10.13.js
- Create 3 scenes in PlugnCast G2.
  - "**playoutSync1**"
  - "**playoutSync2**"
  - "**playoutSync2**"
  - Import the videos in the PlugnCast G2 library and drop it in the scene according to

example	Scene
video_tryptic1.mp4	playoutSync1
video_tryptic2.mp4	playoutSync2
video_tryptic3.mp4	playoutSync2

  - Apply a media behaviour "played infinitely" for all the video media inserted
- Create 3 new video channels and add respectively the scene according to
  - "**playoutSync1**"
  - "**playoutSync2**"
  - "**playoutSync2**"
  - For all the channels, divide the day into 2 sequences:
    - Sequence1 from 8h00 AM to 2h00 PM (6 hours max)
      The 3 channels needs to start at the same time to be synchronized !
    - Sequence2 from 2h00 PM to 8h00 PM (6 hours max)
      The 3 channels needs to start at the same time to be synchronized !
    - Ensure that no media is inserted in calendar from 0h00 AM to 8h00 AM (letting time for the player to reach accurate clock)
- Create 1 new audio channel "AudioPlayoutSync"
  - Add nothing in the audio channel (let empty)
- Create a task "player reboot" every days at 0h00, for your 3 players
  - Add all your players SMA300
- Create 3 new task to upgrade your 3 players
  - "upgrade content player1"
  - "upgrade content player2"
  - "upgrade content player3"
  - For each task, set
    - Video channel "playoutSync<i>"
    - Audio channel "AudioPlayoutSync"
    - Apply a recurrence (planning / every day)
    - Add all your players SMA300



- For each task, in tab "personalization xpath",
  - add a new "**personalization xpath**" permitting to execute the script when it is joined to your playout (adjust the path according to)

```
//xpf:player
```

```
<script src=".medias/Script/xpfSyncManager-V1.10.11.js" xmlns="http://www.innes.fr/2007/XPF10/Language"/>
```

Attention: do use " character (and not MS Windows one ") for Javascript