# SOC JS Playback Engine – RenderingController – Rendering Services - High Level Specifications

V.1.02

Author: marius alexandru

Contents

[SOC JS Playback Engine – RenderingController – Rendering Services - High Level Specifications 1](#_Toc57115383)

[I. Rendering Controller – General Scope 2](#_Toc57115384)

[II. Rendering Services – General Scope 2](#_Toc57115385)

[III. Rendering Services – Class Hierarchy and Class Interfaces 3](#_Toc57115386)

[III.A. Rendering Services – Class Hierarchy and Class Interfaces - Diagram 3](#_Toc57115387)

[III.B. Rendering Services – Class Hierarchy and Class Interfaces – Description 3](#_Toc57115388)

[IV. Rendering Controller – Class Diagram 6](#_Toc57115389)

[IV.A. Rendering Controller – Class Diagram – Diagram 6](#_Toc57115390)

[IV.B. Rendering Controller – Class Diagram – Description 6](#_Toc57115391)

# Rendering Controller – General Scope

The **“Rendering Controller”** **Service** - is the service that controls the Rendering Services based on the commands that it received from the PlaylistController.

This is one of the CRITICAL Components of the application.

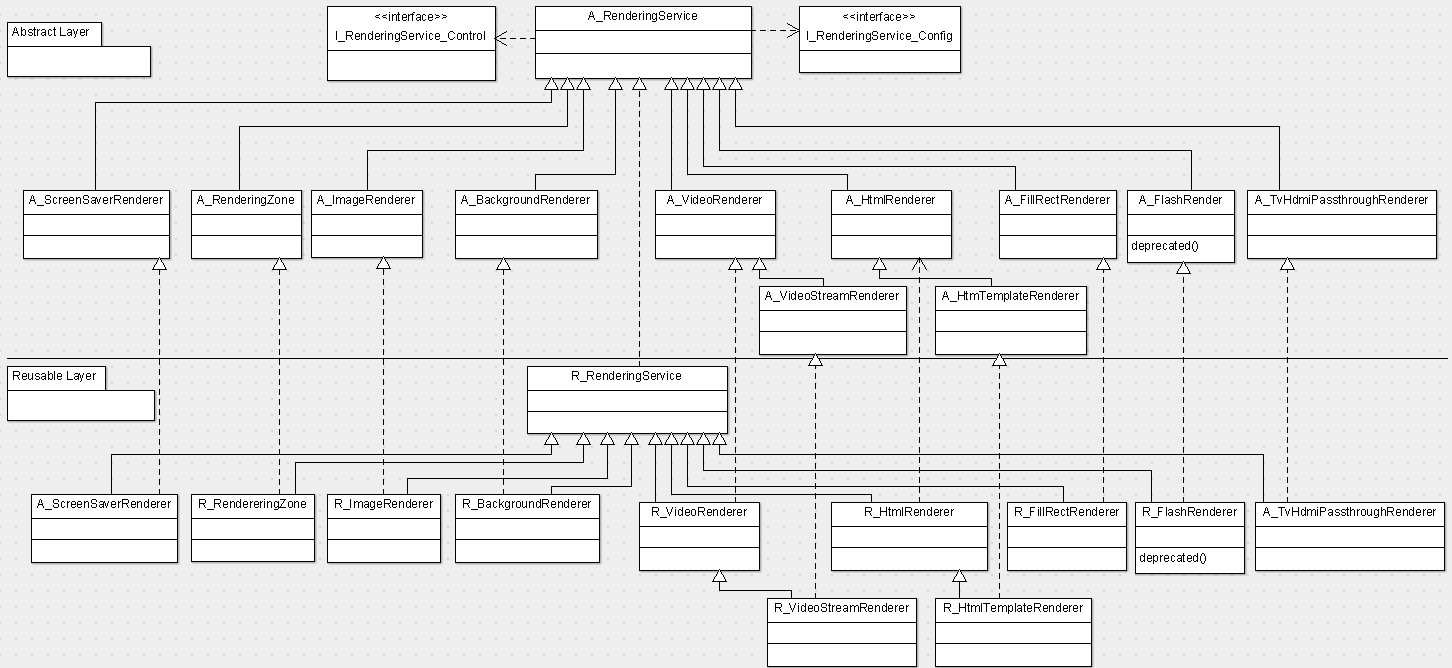
# Rendering Services – General Scope

**The Rendering Services -**

The rendering services are a hierarchy of services that are able to render on the screen different media types. ( A future audio playback will be considered as a rendering service too).

# Rendering Services – Class Hierarchy and Class Interfaces

## III.A. Rendering Services – Class Hierarchy and Class Interfaces - Diagram



## III.B. Rendering Services – Class Hierarchy and Class Interfaces – Description

1. A\_RenderingService - The Renderer Service is the root service of this hierarchy. The service is needed in order to control the rendering services in a uniform way.

It has two interfaces:

I\_RenderingService\_Config – is the interface that supplies functionalities to configure the Rendering Service

I\_RenderingService\_Control – is the interface that supplies functionalities to control the Rendering.

R\_RenderingService - The R\_RenderingService is the reusable realization of A\_RenderingService

2. A\_ImageRenderer – The “Image Renderer Service” perform the rendering of different types of images based on different parameters (stretch, fit, fill, center, normal)

R\_ImageRenderer - The R\_ImageService is the reusable realization of A\_ImageRenderer

3. A\_VideoRenderer – The Video Renderer Service performs the rendering of different types of videos based on different setup parameters (like : keep aspect ratio etc).

R\_VideoRenderer - The R\_VideoRenderer is the reusable realization of A\_VideoRendering

Note: Usually each TV Signage OS supplies a native api for video rendering (that also evolved during different firmwares lifecycles).

For this reason we are talking, about a hierarchy of video renderers services: the Html5VideoRenderer (general available), plus different video renderers supplied by each TV OS that will also have their versions and their evolutions) . The Native Video Renderers will be implemented in v1.1 and v2.0 versions of the application.

4. A\_VideoStreamRenderer – The Video Stream Renderer Service performs the rendering of different types of videos streams. Is an extension of A\_VideoRenderer.

R\_VideoStreamRenderer - The R\_VideoStreamRenderer is the reusable realization of A\_VideoStreamRenderer. This reusable version will work only on the OS that support it. Additional native video stream renderers will be implemented for different OS systems.

5. A\_HtmlRenderer – The HtmlRenderer Service performs the rendering of different types of html pages.

R\_HtmlRenderer - The R\_HtmlRenderer is the reusable realization of A\_HtmlRendering

6. A\_HtmlTemplateRenderer – The Html Template Renderer Service performs the rendering of Mood Html Templates. Is an extension of A\_HtmlTemplateRenderer.

R\_HtmlTemplateRenderer - The R\_HtmlTemplateRenderer is the reusable realization of A\_HtmlTemplateRendering

7. (deprecated) A\_FlashRenderer – The Flash Renderer service performs the rendering of Flash media type. The Flash rendering is not supported by different TV OS, so this type of renderer can be absent on some OS TV playback engines.

R\_ FlashRenderer - The R\_ FlashRenderer is the reusable realization of A\_FlashRenderer. Currently the Flash Player is deprecated, so this renderer will not be implemented.

8. A\_RenderingZone - The RenderingZone service is a rendering service that manage the rendering of a Zone on the Screen. His existence is important in order to manage the positions and the z-order on the screen of different rendering services

R\_RenderingZone - The R\_RendereringZone is the reusable realization of A\_RenderingZone

9. A\_ BackgroundRenderer - The “Background Renderer” Service is a rendering service that manages the display of the playback background (on one or multiple screens)

R\_BackgroundRenderer - The R\_BackgroundRenderer is the reusable realization of A\_ BackgroundRenderer

10. A\_ScreenSaverRenderer – The “Screen Saver Renderer” Service is a rendering service that manages the display of a “screen-saver” when there are no valid media types to be played via the current playlist. Its implementation is based on the implementation of the other rendering services (The Image Renderer and the Video Renderer)

R\_ScreenSaverRenderer - The R\_ ScreenSaverRenderer is the reusable realization of A\_ ScreenSaverRenderer

11. A\_TvHdmiPassThroughService - The HDMI Pass-Through service passes video and audio signals from a high definition (HD) source such as a Blu-ray player or a HD set top box to a home theatre system (HTS) via a HDMI cable. The home theatre system plays the audio from the HD source through its own speakers and sends the unaltered video signal to a TV via a second HDMI cable. Alternatively the original source can be a signal TV from a high definition (HD) source having a TV Tuner.

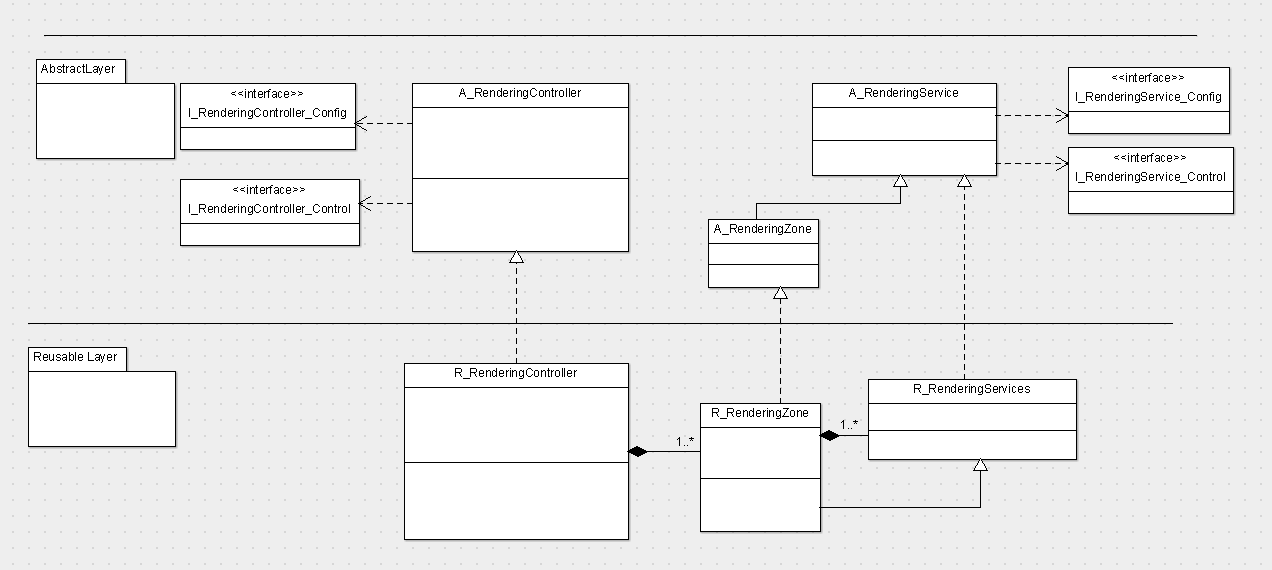
A home theatre system that supports HDMI pass-through will have one HDMI OUT port to send the HDMI signal to a TV, and at least one HDMI IN port to receive a HDMI signal from the connected external device.

Note: The implementation was scheduled for the version 2.0. To discuss if its implementation will be moved earlier.

R\_TvHdmiPassThroughService - The R\_TvHdmiPassThroughService is the reusable realization of A\_ TvHdmiPassThroughService

# Rendering Controller – Class Diagram

## IV.A. Rendering Controller – Class Diagram – Diagram



## IV.B. Rendering Controller – Class Diagram – Description

**A\_RenderingController** - The “Rendering Controller” Service - is the service that controls the Rendering Services.

It has two interfaces:

**I\_RenderingController\_Config** – is the interface that supplies functionalities to configure the Rendering Controller Service

**I\_RenderingController\_Control** – is the interface that supplies functionalities to control different types of Rendering Services.

The **R\_RenderingService** - The R\_RenderingService is the reusable realization of A\_RenderingService

A\_RenderingController Service is composed of 1..n RenderingZones Services

Each **RenderingZone Service** of the RenderingController Service is responsible to manage the rendering inside a Zone on the Screen.

Each RenderingZone Service is composed of 1..n **Rendering Services** of different types.

Based on the media types that should be displayed at one moment inside a Rendering Zone the RenderingZone Service will delegate the rendering of that zone, to a Rendering Service able to render that media type (like ImageRenderer, VideoRenderer, HtmlRenderer etc…)