Design Document of Image Layer for TV

Goals:

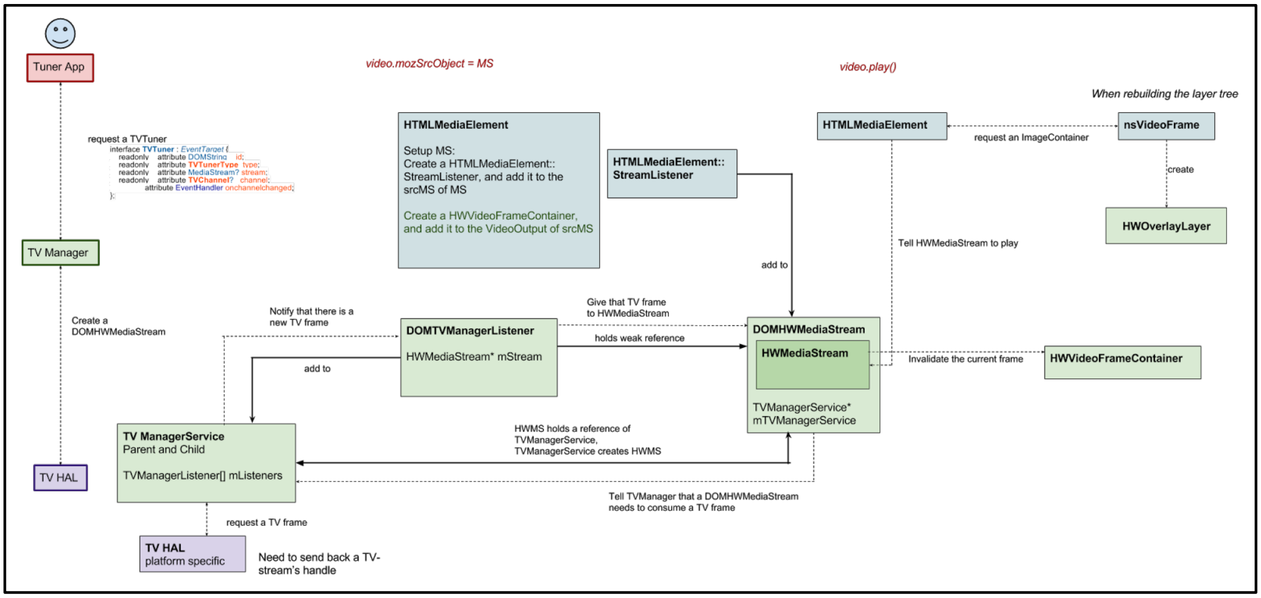
Providing a new layer type for those platforms use a hardware overlay(a component which receives video input directly, instead of receiving decoded data, and overlays with other RGB layers to the screen) to display TV video with UI.

Background knowledge:

1. The BSP and developing environment
   The vendor uses an Posix compliant OS(not linux). Android's HAL and Gonk on Android are also absent in the developing environment. Currently it only uses CompositorOGL to do all the composition jobs.

2. There are two devices that are related to the TV streaming: TV Tuner and TV player. The TV Tuner selects TV channel and video input. And the TV player performs the overlaying and scaling jobs. Here we only need to deal with TV Player device.

3. Components of TV



```
interface TVTuner : EventTarget {
    readonly    attribute DOMString    id;
    readonly    attribute TVTunerType  type;
    readonly    attribute MediaStream? stream;
    readonly    attribute TVChannel?   channel;
                attribute EventHandler onchannelchanged;
};
```

Tuner App
video.mozSrcObject = MS
video.play()
When rebuilding the layer tree
request a TVTuner
HTMLMediaElement
Setup MS:
Create a HTMLMediaElement::StreamListener, and add it to the srcMS of MS
Create a HWVideoFrameContainer, and add it to the VideoOutput of srcMS
HTMLMediaElement::StreamListener
HTMLMediaElement
request an ImageContainer
nsVideoFrame
create
HWOverlayLayer
Tell HWMediaStream to play
add to
TV Manager
Create a DOMHWMediaStream
Notify that there is a new TV frame
DOMTVManagerListener
HWMediaStream* mStream
Give that TV frame to HWMediaStream
holds weak reference
DOMHWMediaStream
HWMediaStream
TVManagerService* mTVManagerService
Invalidate the current frame
HWVideoFrameContainer
add to
TV ManagerService
Parent and Child
TVManagerListener[] mListeners
HWMS holds a reference of TVManagerService, TVManagerService creates HWMS
Tell TVManager that a DOMHWMediaStream needs to consume a TV frame
TV HAL
request a TV frame
TV HAL
platform specific
Need to send back a TV-stream's handle

The streaming of TV is similar to the previewing of camera from MediaStream's aspect. The video can only be started and stopped, but seeking. As the description in TV Media Structure (https://docs.google.com/a/mozilla.com/drawings/d/

1xa5vWmWZejTYHAORkLvn4VsWwt12H0FMQH6IsKKITGw/edit), the relationship between HWOverlay, nsVideoFrame, HTMLMediaElement and HWMediaStream are illustrated in above figure. HWMediaStream is the MediaStream class dedicated for TV, and HTMLMediaElement will be created when a TV VideoTag is presented. The corresponding bug is https://bugzilla.mozilla.org/show_bug.cgi?id=987498 .

Requirements:

1. A video tag will be used to present a TV screen, and then a an image layer with new type will also be created in layer tree for composition. The TV screen could be presented as full screen or in a relative small window, and the input may be one of following source: tuner, HDMI, camera, and recorded video file. Due to the Image Layer for TV has no actual pixel data, we need to introduce a new kind of image class - OverlayImage, to provide a placeholder for Image Layer of TV input, which only contain size, position, and handles relative to TV manager.

2. Unlike the image updating process of Camera preview - whenever VideoFrameContainer:: SetCurrentFrame() is called, the OverlayImage is not always need to be updated. It would be only updated whenever there are changes relative to geometry or input changes of TV, such as:
a. TV window resizing,
b. Input changes: i.e. from TV to HDMI,
c. Two inputs are presented in the same time, such as PIP(Picture in picture), POP(picture outside picture), and PAP(picture and picture) mode.

3. While the Image Layer gets composed, CompositorOGL should make a transparent area for TV layer in the RGBA buffer(where UI layers are composed). Due to the hardware overlay places the RGB(UI) layers above the Image Layer of TV, the transparent area is used in order to let TV layer be shown through the specified area.

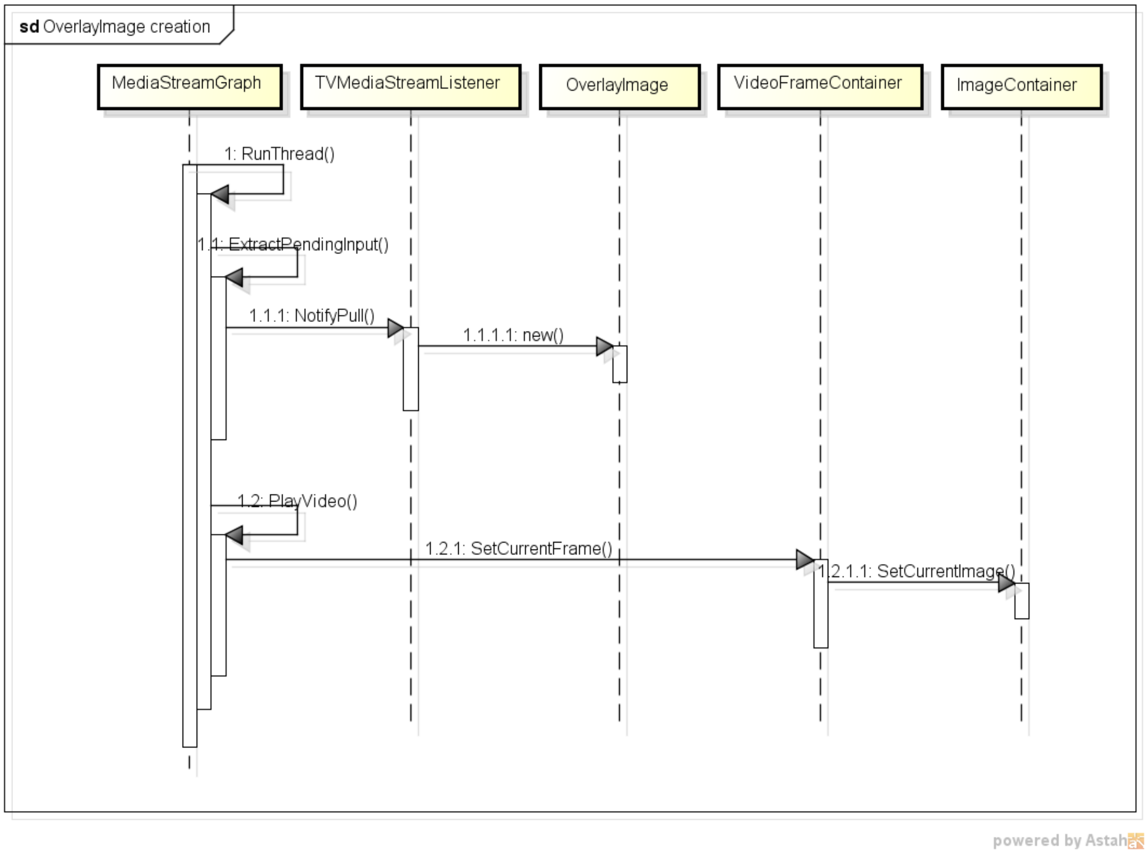
4. Introduce HwcComposer2D to this TV project to overlay UI/RGB layer and Image Layer of TV. Setting size and position of TV input should be performed in corresponding HAL functions, which be called by the composition function of HwcComposer2D.

High Level Design

1. OverlayImage

a. Initialization

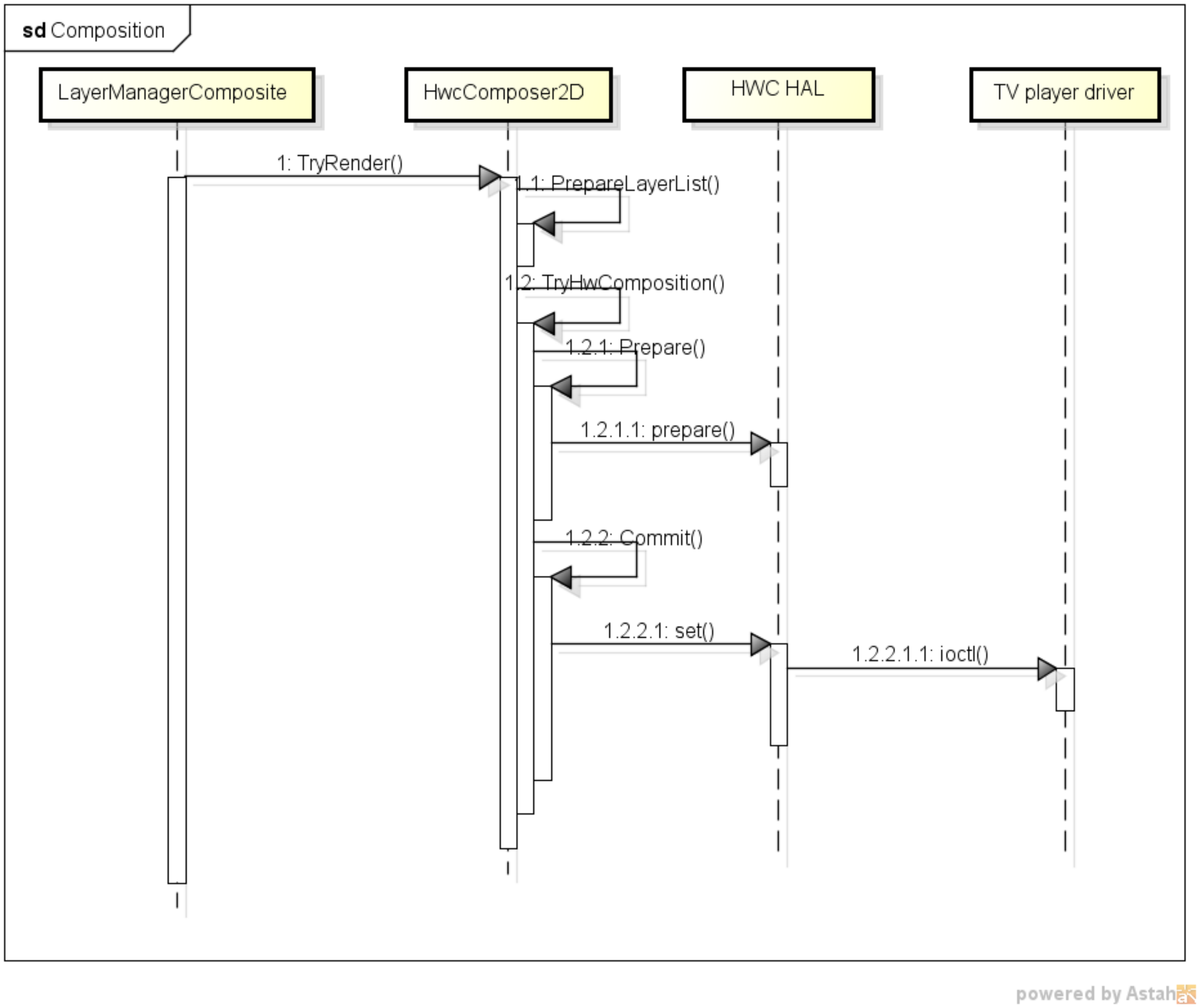
From the design of bug#987498, there is a dedicated stream listener for TV - TVMediaStreamListener. The OverlayImage class should be created by TVMediaStreamListener, and then MediaStreamGraphImpl will pass the image to ImageContainer through SetCurrentFrame(). OverlayImage should only be created once per TV input, not every time when SetCurrentFrame() is called(which is the case of camera previewing).



b. Composition

While compositing, a transparent area which correspond to the layer of TV screen needs to be made, to let let TV image be shown through the specified area. ImageLayerComposite should prepare an specific effect chain in RenderLayer(), and then CompositorOGL can use this hint to punch a rectangle hole in DrawQuad().

2. HwcComposer2D
By introducing HwcComposer2D , we can perform the the overlay function through Commit(). As the design in Android gonk ,the vendor should implement driver manipulation job in the HWC HAL, such that HwcComposer2D can calls corresponding HAL functions - set(), prepare(), and query().


sd Composition
LayerManagerComposite
HwcComposer2D
HWC HAL
TV player driver
1: TryRender()
1.1: PrepareLayerList()
1.2: TryHwComposition()
1.2.1: Prepare()
1.2.1.1: prepare()
1.2.2: Commit()
1.2.2.1: set()
1.2.2.1.1: ioctl()
powered by Astah