November 2019

Unified management of media sessions and audio focus

Rebecca Hughes

Mounir Lamouri

Unified management of media sessions and audio focus

ABSTRACT

Multiple simultaneous audio sources are typically handled by the media session and audio focus mechanisms of an operating system (OS). These mechanisms typically permit only a single audio source to output sound at any given time. Switching to another component that generates sound results in a switch from the currently playing audio to the audio of the new component which may not always provide an optimal user experience. Moreover, media session and audio focus mechanisms of the different operating systems, e.g., a guest OS running as a container or virtual machine atop a host OS may not interoperate in a seamless manner, thus making it difficult to provide a unified media experience. This disclosure describes mechanisms to handle multiple audio sources that request sound output at the same time. A single media session and audio focus service is provided that handles audio output requests from all audio sources. The service is designed to aggregate the various media sessions and apply appropriate rules to determine which of the multiple simultaneous requests for audio playback are played at any given time.

KEYWORDS

- Media session
- Audio focus
- Simultaneous audio
- Virtual machine (VM)
- Operating system
- Guest OS

BACKGROUND

Multiple sources of audio on a device, e.g., a computer, a tablet, etc. can generate sound output at the same time. These sources can include: the device operating system (OS), an application, subcomponents of an application such as tabs within a web browser, a voice assistant, a guest OS running in a container or as a virtual machine, etc. Multiple simultaneous audio sources are typically handled by the media session and audio focus mechanisms of an operating system. The audio focus mechanisms permit only a single audio source to output sound at any given time. However, multiple audio sources are allowed to play in some cases, such as a notification announcement played over a song being played in a music application. The management of audio focus is aided by media session mechanisms that provide metadata about each audio source. For instance, the metadata for the audio of a song can include the track title, artist, album image, length, genre, etc.

However, the media session and audio focus mechanisms are typically able to handle multiple audio sources in a limited way, such as being restricted to the audio generated by various browser tabs but not to other applications or OS components. Moreover, in the presence of a guest OS, the media session and audio focus mechanisms of the different operating systems may not interoperate in a seamless manner, thus making it difficult to provide a unified media experience.

Since audio focus mechanisms generally restrict sound output to only one media source, switching to a component that generates sound results in a switch from the currently playing audio to the audio output of the new component. Such a switch may not always provide an optimal user experience (UX). The issue can be avoided by having a single media session for all

components of an application or all applications of an OS. However, such an approach results in the loss of metadata and playback control of individual audio sources.

DESCRIPTION

This disclosure describes mechanisms to handle multiple audio sources that request sound output at the same time on a device such as a computer, a tablet, a smart speaker, etc. To this end, an operating system is provided that includes a single media session and audio focus service that handles audio output requests from all of the various audio sources, such as OS components, application components, other OSes running in VMs, etc. A request for audio focus is relayed to the service whenever a new media session emerges. The service is designed to aggregate the various media sessions within the OS and apply appropriate rules to determine which of the multiple simultaneous requests for audio playback should be played at a given time.

Each application contains its own media session and audio focus mechanisms that operate in conjunction with the media session and audio focus service of the OS. Further, the various media sessions connected to the respective audio sources within a single application are grouped together, and the group is assigned a group identifier (groupID). For example, the media sessions of audio requests by the various individual tabs within a web browser window can be grouped under a single groupID for that browser window as a whole. The group with which a given media session is associated can change as appropriate. For instance, if a web browser tab is moved from one window to another, the media session associated with that tab is correspondingly moved to the respective media session group of the other window.

If any of the media sessions within a group is granted audio focus, all other media sessions within the same group share the audio focus with that media session. As a result, a

request for audio focus by another media session within the same group does not suspend the ongoing audio playback of the media session that was granted focus earlier.

A guest OS operating as a VM or in a container can have its own media session and audio focus service. Audio requests from such OSes are tracked via a translation layer to convert the corresponding media sessions into a native format of the host OS. The translation mechanisms permit audio requests from applications and components within the VM or a container to operate similar to native applications on the host OS.

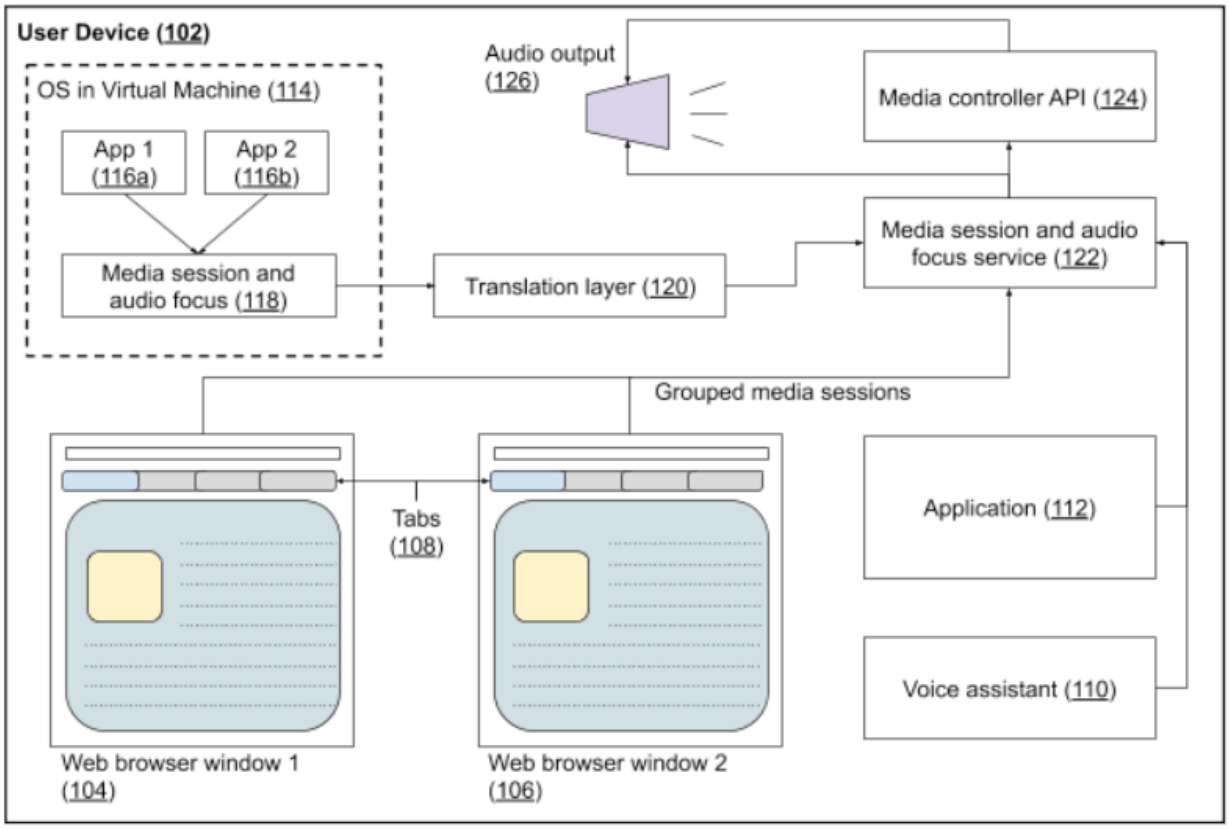

Fig. 1: Audio output determined by unified management of media sessions and audio focus

Fig. 1 shows an operational example of the techniques described in this disclosure. A user device (102) includes multiple sources of audio: tabs (108) within two web browser windows (104 and 106), a voice assistant (110), an application (112), and an OS running within a VM

(114). A single media session and audio focus service (112) on the user device is used to process the audio requests from the media sessions of the various audio sources.

The media sessions of the individual tabs within each browser window are grouped together prior to being passed on to the media session and audio focus service. The OS within the VM provides is own media session and audio focus mechanisms (118) to the apps (116a, 116b) running on the OS. The output of the media session and audio focus mechanisms of the OS within the VM is processed via a translation layer (120) and converted to the appropriate format for the media session and audio focus service of the native OS of the user device. Unified processing of all received media sessions and/or media session groups is used to determine the appropriate audio output (126). Alternatively, or in addition, the audio output can be controlled via a media controller API (124) that provides access to the media session and audio focus mechanisms.

The media controller application programming interface (API) is provided to enable access to the media session and audio focus service of the OS. The media controller API serves as the single point for controlling audio playback and retrieving metadata of media sessions. The API can be leveraged to provide seamless user interfaces (UIs) and UX for functionalities that involve audio playback. For example, a notification can surface controls for each active sound source. Similarly, appropriate audio-related actions can result from corresponding user actions. For instance, all media playback can be paused when a headset is disconnected.

The techniques described in this disclosure can be applied to improve the media playback UI/UX on any device or OS that needs to handle multiple simultaneous requests for audio playback.

CONCLUSION

This disclosure describes mechanisms to handle multiple audio sources that request sound output at the same time. A single media session and audio focus service is provided that handles audio output requests from all audio sources. The service is designed to aggregate the various media sessions and apply appropriate rules to determine which of the multiple simultaneous requests for audio playback are played at any given time. A media controller API is provided to access the service and control audio playback and retrieve metadata of media sessions. Each application contains its own media session and audio focus mechanisms that operate in conjunction with the media session and audio focus service of the OS. Further, the various media sessions connected to the respective audio sources within a single application are grouped. If any of the media sessions within a group is granted audio focus, then all other media sessions within the same group share the audio focus with that media session. A translation mechanisms permits audio requests from applications and components within an OS running in a VM or a container to operate similar to native applications on the host OS.

REFERENCES

1. Media Session Standard, available online at https://w3c.github.io/mediasession/
2. Brinkmann, Martin, “Chrome: new audio focus feature launches” available online at https://www.ghacks.net/2016/11/11/chrome-new-audio-focus-feature-launches/