USB and content camera

# Device category and release timeline

|  |  |  |  |
| --- | --- | --- | --- |
| Device Category | Required (Y/N) | Feature release targeted for | Shared with the following Partners: |
| Teams Phones (desk phone) | No | **N/A** | **N/A** |
| Teams Phones (conference phone) | No | **N/A** | **N/A** |
| Teams Phones (Low Cost Phone) | No | **N/A** | **N/A** |
| Teams Displays | No | **N/A** | **N/A** |
| Teams Rooms on Android | Yes | **2022 FW Update #1** | **Lenovo , Yealink, Crestron, Poly, Audiocodes, Jabra, Logitech, EPOS, Neat** |
| Teams Panels | No | **N/A** | **N/A** |

# Overview

This feature is based on android camera2 APIs. The firmware must expose a USB camera connected and disconnected state using standard Android Camera2 APIs. Teams app should be able to use camera2 APIs to find, query capabilities and read frames.

Android camera2 guide: https://developer.android.com/training/camera2

# Goals

1. Transition to camera2 APIs on all MTRA devices
2. Report peripheral cameras to MTRA devices with proper information
3. Identify content cameras that cannot be used as a room camera.

# Non-Goals

1. Change the UI/UX of MTRA devices when it comes to camera
2. Select the default camera
3. Do video format conversion for cameras in Microsoft Teams.

# Functional Specification

## New characteristics reported by firmware

Firmware reports custom characteristics through *android.hardware.camera2. CameraCharacteristics.Key<String>.* The newly supported characteristics are as follows:

|  |  |  |
| --- | --- | --- |
| **Key name** | **Value type** | **Value** |
| teams.camera.name | String | Readable name of a USB camera |
| teams.camera.manufacturer | String | Manufacturer of a USB camera |
| teams.camera.model | String | Model of a USB camera |
| teams.camera.pid | String | Camera UVC device product ID |
| teams.camera.vid | String | Camera UVC device vendor ID |

## USB camera lens facing

The lens facing of a USB camera should be LENS\_FACING\_EXTERNAL which reported by the existing *android.hardware.camera2. CameraCharacteristics.* *LENS\_FACING.*

## Video format of USB camera

A USB camera must send RAW video stream(NV21 format) to Teams app:

* If OEM chooses USB 2.0, OEMs must convert MJPEG to RAW data and expose NV21 format to Teams;
* If OEM chooses USB 3.0 and RAW data is directly received from camera, no conversion is needed.

## Content camera

Any USB camera can be selected as a content camera in Teams settings if there are two or more cameras available and at least one of them is a UVC camera.

And OEMs can choose to designate a USB camera for content camera purposes only, in this case this camera will not be selectable as a room video camera by Teams app.

Designate a USB camera as content camera only

If a USB camera is designated as a content camera only. Then it must be reported with a camera characteristic representing the camera type. Teams client will check the camera type of the available camera using the code below to ensure the camera is a content camera only:

Camera characteristic name: **camera.type**

Value of camera type: **external\_content\_camera**

*CameraManager cameraManager = (CameraManager) context.getSystemService(Context.CAMERA\_SERVICE);*

*cameraManager.registerAvailabilityCallback(****new*** *CameraManager.AvailabilityCallback() {*

***@Override***

***public******void******onCameraAvailable****(****@NonNull*** *String cameraId) {*

***super****.onCameraAvailable(cameraId);*

***try*** *{*

*CameraCharacteristics characteristics = cameraManager.getCameraCharacteristics(cameraId);*

*String cameraType = characteristics.get(cameraTypeCharacteristic);*

***if*** *(cameraType !=* ***null*** *&& cameraType.equals("external\_content\_camera")) {*

*// Content camera detected, start screen sharing session.*

*}*

*}* ***catch*** *(Exception ex) {*

*// handle exception*

*}*

*}*

***@Override***

***public******void******onCameraUnavailable****(****@NonNull*** *String cameraId) {*

***super****.onCameraUnavailable(cameraId);*

*// if content camera disconnected, stop any active screen sharing session.*

*}*

*},* ***new*** *Handler());*

Please note that this designated camera ability is not necessarily required for devices and cameras from different manufactures.