

Metadata and Controls

Metadata support

To support the saving of raw image files by the Android framework, substantial metadata is required about the sensor's characteristics. This includes information such as color spaces and lens shading functions.

Most of this information is a static property of the camera subsystem and can therefore be queried before configuring any output pipelines or submitting any requests. The new camera APIs greatly expand the information provided by the `getCameraInfo()` method to provide this information to the application.

In addition, manual control of the camera subsystem requires feedback from the assorted devices about their current state, and the actual parameters used in capturing a given frame. The actual values of the controls (exposure time, frame duration, and sensitivity) as actually used by the hardware must be included in the output metadata. This is essential so that applications know when either clamping or rounding took place, and so that the application can compensate for the real settings used for image capture.

For example, if an application sets frame duration to 0 in a request, the HAL must clamp the frame duration to the real minimum frame duration for that request, and report that clamped minimum duration in the output result metadata.

So if an application needs to implement a custom 3A routine (for example, to properly meter for an HDR burst), it needs to know the settings used to capture the latest set of results it has received in order to update the settings for the next request. Therefore, the new camera API adds a substantial amount of dynamic metadata to each captured frame. This includes the requested and actual parameters used for the capture, as well as additional per-frame metadata such as timestamps and statistics generator output.

Per-setting control

For most settings, the expectation is that they can be changed every frame, without introducing significant stutter or delay to the output frame stream. Ideally, the output frame rate should solely be controlled by the capture request's frame duration field, and be independent of any changes to processing blocks' configuration. In reality, some specific controls are known to be slow to change; these include the output resolution and output format of the camera pipeline, as well as controls that affect physical devices, such as lens focus distance. The exact requirements for each control set are detailed later.

Raw sensor data support

In addition to the pixel formats supported by the old API, the new API adds a requirement for support for raw sensor data (Bayer RAW), both for advanced camera applications as well as to support raw image files.

IN THIS DOCUMENT

[Metadata support](#)

[Per-setting control](#)

[Raw sensor data support](#)