ETSI TS 126 119 V18.0.0 (2024-05)

TECHNICAL SPECIFICATION

LTE;
5G;
Media Capabilities for Augmented Reality
(3GPP TS 26.119 version 18.0.0 Release 18)

A GLOBAL INITIATIVE

Reference

DTS/TSGS-0426119vi00

Keywords

5G,LTE

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from:
https://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

If you find a security vulnerability in the present document, please report it through our
Coordinated Vulnerability Disclosure Program:
https://www.etsi.org/standards/coordinated-vulnerability-disclosure

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.
No recommendation as to products and services or vendors is made or should be implied.
No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.
In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found under https://webapp.etsi.org/key/queryform.asp.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

shall indicates a mandatory requirement to do something

shall not indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

should indicates a recommendation to do something

should not indicates a recommendation not to do something

may indicates permission to do something

need not indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

can indicates that something is possible

cannot indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

will indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

will not indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

might indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

might not indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

is (or any other verb in the indicative mood) indicates a statement of fact

is not (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

Introduction

The present document provides technologies for the deployment of Augmented Reality (AR) as defined in 3GPP TR 26.928 [2] and the execution of Augmented Reality applications on targeted devices such as those identified in 3GPP TR 26.998 [3].

On the spectrum of eXtended Reality (XR) experiences, Augmented Reality overlay virtual information on top of the user's perception of the real environment. Those virtual and real components of the scene seamlessly blend together from the user's perspective. Additionally, some AR experiences can enable interactivity between the user and the virtual components of the scene.

In the present document, the focus lies in the definition of the media capabilities for AR devices, including media format encapsulation capabilities, media codec capabilities, processing function capabilities. The related minimum required performances for different device types are also defined.

1 Scope

The present document defines the supported media formats, codecs, processing functions for XR Devices in UE per XR device type category. The present document addresses the interoperability gaps identified in the conclusions of TR 26.998 [3].

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TR 26.928: "Extended Reality (XR) in 5G".

[3] 3GPP TR 26.998: "Support of 5G glass-type Augmented Reality / Mixed Reality (AR/MR) devices".

[4] 3GPP TR 26.857: "5G Media Service Enablers".

[5] Khronos, "The OpenXR Specification", https://registry.khronos.org/OpenXR/specs/1.0/html/xrspec.html.

[6] 3GPP TS 26.506: "5G Real-time Media Communication Architecture (Stage 2)".

[7] ITU-T Recommendation H.264 (08/2021): "Advanced video coding for generic audiovisual services".

[8] ITU-T Recommendation H.265 (08/2021): "High efficiency video coding".

[9] 3GPP TS 26.117: "5G Media Streaming (5GMS); Speech and audio profiles".

[10] ISO/IEC 12113:2022 Information technology Runtime 3D asset delivery format Khronos glTF™2.0

[11] ISO/IEC 23090-14:2023 Information technology Coded representation of immersive media Part 14: Scene description

[12] ISO/IEC 23090-14:2023/Amd 1:2023 Information technology Coded representation of immersive media Part 14: Scene description

[13] ISO/IEC 23090-14:2023/DAmd 2 Information technology Coded representation of immersive media Part 14: Scene description

3 Definitions of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

Anchor: a virtual element for which its position, orientation, scale and other properties are expressed in the trackable space defined by the trackable. A virtual asset's position, orientation, scale and other properties are expressed in relation to an anchor.

Media Session Handler: a set of functions responsible for handling all 5G control plane operations, such as requesting network assistance, discovering and allocating edge resources, etc.

Presentation Engine: a set of composite renderers, rendering the component of the scenes.

Reference Points: geometric points whose position is identified both mathematically and physically.

Trackable: a real-world object that can be tracked by the XR runtime. Each trackable provides a local reference space, also known as a trackable space, in which an anchor can be expressed.

Swapchain: a queue of images shared between the XR Application and the XR Runtime

Swapchain image: image in a swapchain.

XR Application: application running on an XR Device which offers an XR experience based on an XR Runtime.

XR Device: a device capable of offering an XR experience.

XR Runtime: Set of functions provided by the XR Device to the XR Application to create XR experiences.

XR Runtime API: the API to communicate with an XR Runtime.

XR Scene Manager: a set of functions that supports the application in arranging the logical and spatial representations.

XR Session: an application's intention to present XR content to the user.

XR Source Management: management of data sources provided through the XR runtime.

XR System: a collection of resources and capabilities from the XR Runtime exposed to the XR Application for the duration of the XR Session.

XR View: a rendered view of the scene generated by the XR Application and passed on to the XR Runtime during a running XR Session**XR Space**: a frame of reference in which 3D coordinates are expressed.

Warping: correcting the rendered image based on the latest head pose estimation

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

API	Application Programming Interface
AR	Augmented Reality
AAC	Advanced Audio Coding
AVC	Advanced Video Coding
CPB	Coded Picture Buffer
DPB	Decoded Picture Buffer
ELD	Enhanced Low Delay
EVS	Enhanced Voice Services
glTF	graphics library Transmission Format
GLB	glTF Binary
HEVC	High Efficiency Video Coding
HMD	Head-Mounted Display
HRD	Hypothetical Reference Decoder
HSS	Hypothetical Stream Scheduler
IVAS	Immersive Voice and Audio Services
JSON	JavaScript Object Notation
MPEG	Moving Picture Expert Group
MPEG SD	MPEG Scene Description

MR	Mixed Reality
OP	Observation Point
SLAM	Simultaneous Localisation And Mapping
UE	User Equipment
VCL	Video Coding Layer
VR	Virtual Reality
XR	eXtended Reality

4 XR concepts and device types

4.1 XR concepts

4.1.1 General

Extended Reality (XR) refers to a continuum of experiences combine real-a and- virtual combined environments in which the user is immersed through one or more devices capable of audio, visual and haptics rendering generated by computers through human-machine interaction. XR encompasses technologies associated with Virtual Reality (VR), Augmented Reality (AR) and Mixed Reality (MR) which constitute the so-called XR continuum. A detailed overview of definitions, concepts and background on XR and AR is provided in TR 26.928 [2] and TR 26.998 [3], respectively.

The terms Augmented Reality, Virtual Reality, Mixed Reality and eXtended Reality as used throughout this document are defined in Clause 4.1 of 3GPP TR 26.928 [2].

4.1.2 XR Device

An XR device is capable of offering an XR experience. An XR Device is assumed to have one or several displays, speakers, sensors, cameras, microphones, actuators, controllers and/or other peripherals that allow to create XR experiences, i.e. experiences for which the user interacts with the content presented in virtual world and/or augmented to the real-world. Example of XR Devices are AR Glasses, a VR/MR Head-Mounted Display (HMD) or a regular smartphone, etc.

4.1.3 XR application

An application which offers an XR experience by making use of the hardware capabilities, including media capabilities, of the XR Device it runs on as well as the network connectivity to retrieve the asset being used by the application is referred to as an XR Application. In the context of this specification, it is primarily assumed that access to the network is provided by 5G System functionalities.

To enable XR experiences, the hardware on an XR Device typically offers a set of functions to perform commonly required XR operations. These operations include, but are not limited to:

- accessing controller/peripheral state,

- getting current and/or predicted tracking positions and pose information of the user,

- receiving or generating pre-rendered views of the scene for final presentation to the user, taking into account the latest user position and pose. Adaptation to the latest user position and pose is also referred to as warping.

4.1.4 XR Runtime

4.1.4.1 General

XR Runtime provides a set of functionalities to XR applications including but not limited to peripheral management, runtime functions as tracking, SLAM, composition and warping etc. The functions are accessible to the XR Application via an API exposed by the XR Runtime referred to as the XR Runtime Application Programming Interface (XR API). The XR Runtime typically handles functionalities such as composition, peripheral management, tracking, Spatial Localization and Mapping (SLAM), capturing and audio-related functions. Further, it is assumed that the hardware and software capabilities of the XR Device are accessible through well-defined device APIs, and in particular the media capabilities are accessible through media APIs.

An overview of an XR Device logical components is shown in Figure 4.1.4-1.

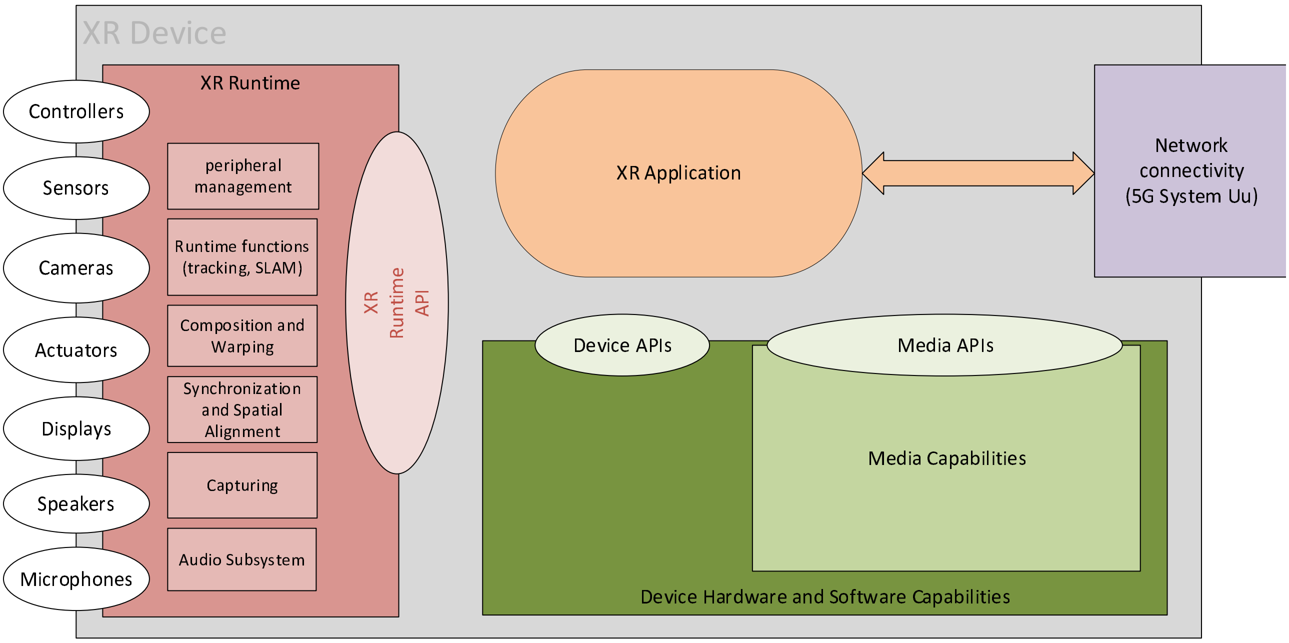

Figure 4.1.4-1 Logical components of an XR Device

This specification relies on a hypothetical XR Runtime and its API in order to define the media capabilities. This way, different implementation of XR runtimes may be compatible with this specification. However, for the purpose of developing this specification, the minimal set of expected functionalities of the XR Runtime has been aligned with the core Khronos' OpenXR specification [5]. Support for other XR Runtime environments is not precluded by this approach. Lastly, a mapping of general functionalities to OpenXR is provided in Annex B.

4.1.4.2 XR session and rendering loop using XR Runtime (informative)

At startup, the XR Application creates an XR Session via the XR Runtime API and allocates the necessary resources from the available resources on the XR Device. Upon success, the XR Runtime begins the life cycle of the XR Session whose cycle is typically made of several states. The purpose of those states is to synchronise the rendering operations controlled by the XR Application with the display operations controlled by the XR Runtime. The rendering loop is thus a task jointly executed by the XR Runtime and the XR Application and synchronised via the states of the XR Session.

The XR Application is responsible of generating a rendered view of the scene from the perspective of the user. To this end, the XR Application produces XR Views which are passed to the XR Runtime at iterations of the rendering loop. The XR Views are generated for one or more poses in the scene for which the XR application can render images. From those views, the view corresponding to the viewer's pose is typically called the primary view. There may be other XR Views defined in the scene, for instance for spectators.

The XR Views are configured based on the display properties of the XR Device. A typical head-mounted XR System has a stereoscopic view configuration, i.e. two views, while a handheld XR Device has a monoscopic view configuration, i.e. a single view. Other view configurations may exist. At the start of session, the XR Application configures the view type based on those device properties which remains the same for the duration of the XR Session.

A XR View may also comprise one more composition layers associated with an image buffer. Those layers are then composed together by the XR Runtime to form the final rendered images.

In addition to layers containing visual data, an XR View may be complemented with a layer provided depth information of the scene associated with this XR View. This additional information may help the XR Runtime to perform pose correction when generating the final display buffer. Another type of layer can be an alpha channel layer useful for blending the XR View with the real environment for video-see through XR devices, e.g. which is the case for AR applications running on smartphones.

For the XR Application to render the XR Views, the XR Runtime provides the viewer pose as well as projection parameters which are typically taken into account by applications to render those different XR Views. The viewer pose and projection parameters are provided for a given display time in the near future. The XR Runtime accepts repeated calls for prediction updates of the pose, which may not necessarily return the same result for the same target display time. Instead, the prediction gets increasingly accurate as the function is called closer to the given time for which a prediction is made. This allows an application to prepare the predicted views early enough to account for the amount of latency in the rendering while at the same time minimising the prediction error when pre-rendering the views.

In addition, the XR Application communicates with input devices in order to collect actions. Actions are created at initialization time and later used to request input device state, create action spaces, or control haptic events. Input action handles represent ‘actions’ that the application is interested in obtaining the state of, not direct input device hardware.

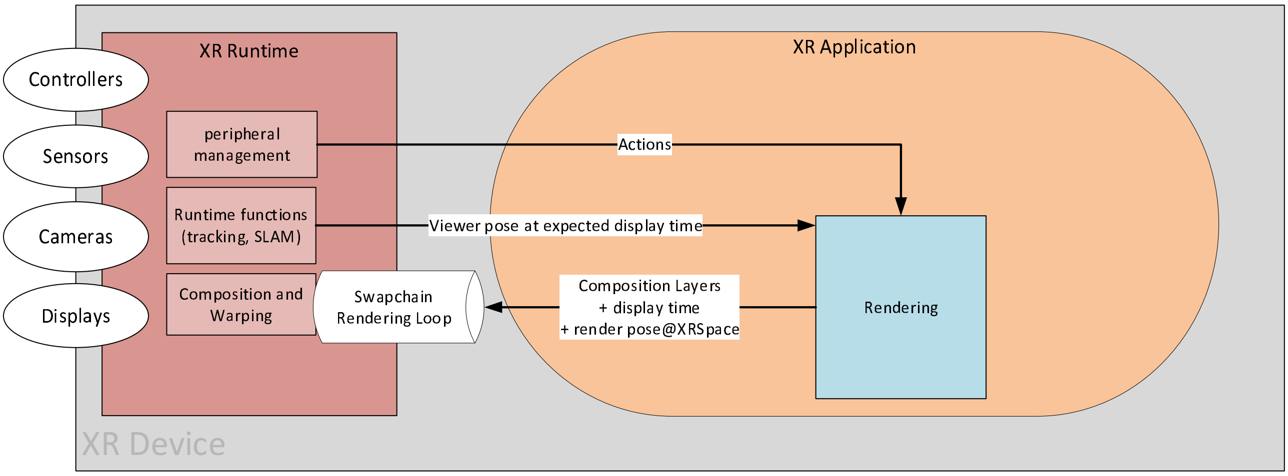

Figure 4.1.4-1 Rendering loop for visual data

4.2 Media pipelines and rendering loop

In the context of this specification, media to be rendered and displayed by the XR Device through the XR Runtime is typically available in an compressed form on the device. Media is accessed using a 5G System, decoded in the device using media capabilities, and then the decoded media is rendered to be provided through swapchains to the XR Runtime as shown in Figure 4.2-1.

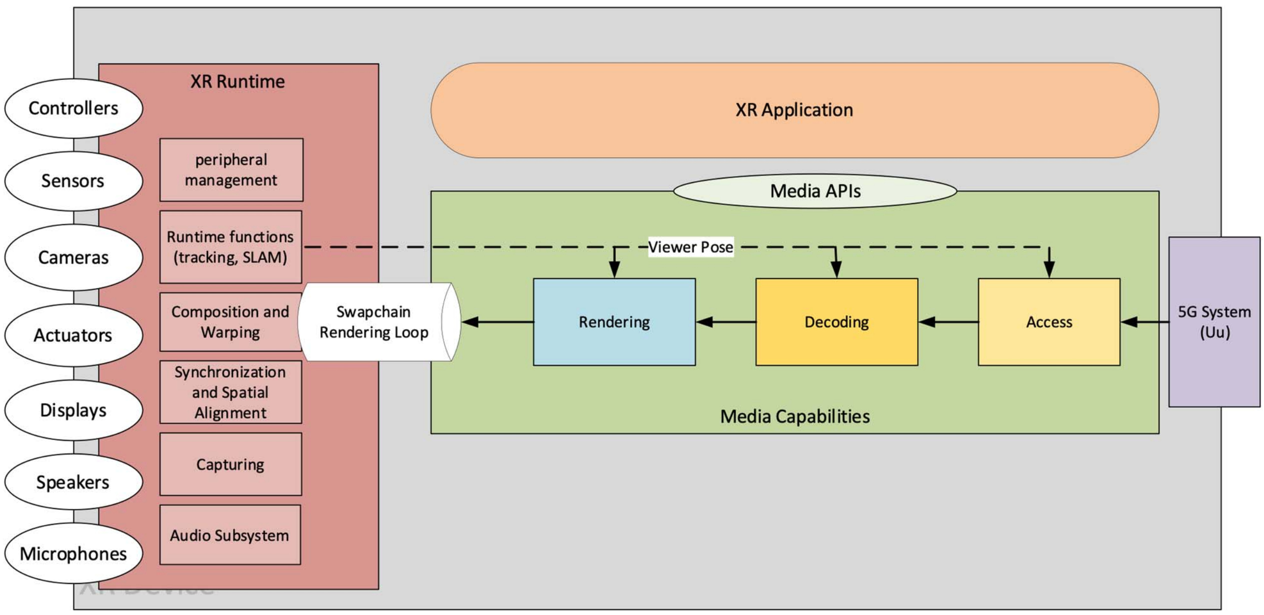

Figure 4.2-1 Media pipelines: Access, decoding and rendering

The rendering function is responsible of generating the content that will be presented by the XR Runtime. This rendering function makes use of rendering loops and provide the results of those loops to the XR Runtime via swapchains. The application sets up different pipeline that comprise processes for media access, decoding, and view rendering. To configure those pipelines and the properties of the generated views (e.g. number of layers, stereoscopic/monoscope views), the rendering function needs to have access to the information about the current session defined at the initialisation step:

- View configuration
- Blend modes
- XR spaces
- swap chain formats and images
- projection layer types

4.3 Device Types

4.3.1 Device type 1: Thin AR glasses

The thin AR glasses device type represents a type of device which is considered as power-constrained and with limited computing power with respect to the other device types. These limitations typically come from the requirement to design a device with a small and lightweight form factor. Regarding rendering capacity, this device type is expected to rely on remote rendering to be able display complex scenes to the user. For example, such device type may run a split rendering session where the split rendering server delivers pre-rendered views of the scene. However, devices in this category can still operate without external support for applications that do not require complex rendering capabilities, for instance, text messaging, 2D video communication, etc. Lastly, the thin AR glasses offers AR experiences to the user via optical see-through display.

4.3.2 Device type 2: AR glasses

The AR glasses device type represents a type of device which is considered to have higher computation power compared to the thin AR glasses device type. As a result, this AR device type has higher rendering capacities and is generally expected to be capable of rendering scenes without external support, even though remote rendering is not precluded to lower the power consumption on the device or enable the display of scenes beyond the device's rendering capability. Lastly, the AR glasses offers AR experiences to the user via optical see-through display.

4.3.3 Device type 3: XR phone

The XR phone device type represents a type of device which corresponds to a smartphone with capacities and resources sufficient to offer AR experiences. As a result, this device type is capable of rendering scenes without external support. Lastly, the XR phone offers AR experiences to the user via video see-through display.

4.3.4 Device type 4: XR Head Mounted Display (HMD)

The XR HMD device type represents a type of device which corresponds to HMDs capable of offering at least AR experiences but not precluding other types of XR experiences. This device type is expected to be capable of rendering scenes without external support. Lastly, the XR phone offers AR experiences to the user via video see-through display.

5 Device reference architecture and interfaces

5.1 Architecture

The XR Baseline Client represents the functionalities, the peripherals, and the interfaces that are present on a generic XR UE. The XR Baseline Client reference architecture is shown in Figure 5.1-1. The actual device may be realized by a single device, or a combination of devices linked together. The details on how to instantiate an XR Baseline Client in the context of a service or deployment scenario is left for the respective Work Items and Study Items to define.