**Agenda item:** 10.5

**Source:** Qualcomm Inc.

**Title: Architectural Considerations for iRTCW**

**Document for** Discussion andAgreement

# Introduction

In this contribution, we discuss some considerations on the architecture for iRTCW.

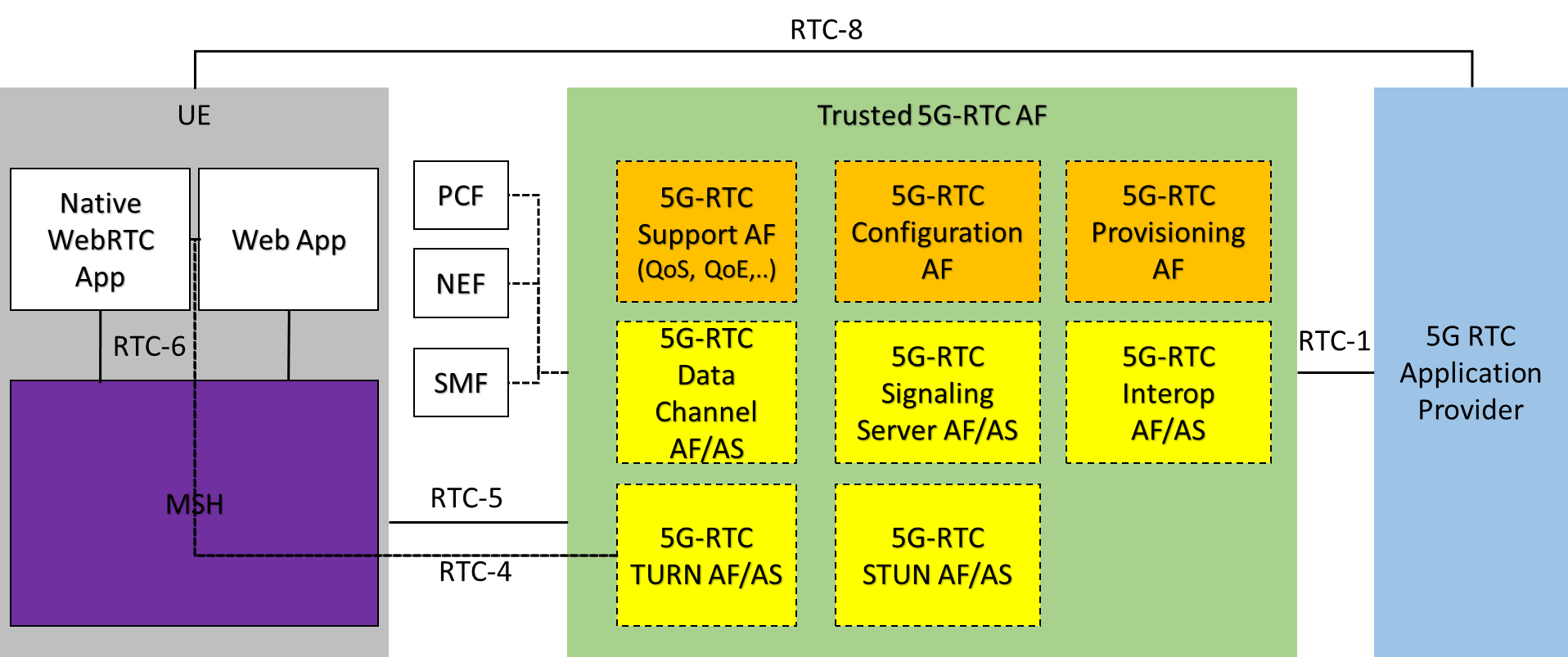
# Architectural Considerations

We have agreed on 4 different collaboration scenarios for iRTCW at the SA4-118e meeting. The collaboration scenarios are listed here for convenience:

1. 5G support for OTT WebRTC
2. MNO-provided trusted WebRTC functions
3. MNO-facilitated WebRTC services
4. Inter-operable WebRTC services

Upon the definition of 5GMS, similar collaboration scenarios have been considered and a 5GMS architecture has been developed to accommodate the needs of these different scenarios. A similar exercise is required for iRTCW. The architecture offers flexibility through a set of functions and interfaces that can be combined in different ways based on the scenario’s needs.

In this contribution, we identify the WebRTC functions that may be used in the different collaboration scenarios. We propose a potential mapping between the WebRTC functions and 5G Application Functions (AF) or Application Servers (AS). We assume that 5G\_AREA will define the architecture for 5G RTC and the resulting architecture will then be adopted by iRTCW for proper mapping. A potential such mapping is depicted in the following diagram and assumed for the rest of the document..



The RTC-1 interface allows the Application Provider to provision support for RTC sessions that are offered by it. The provisioning may cover the following functionalities:

* QoS support for WebRTC sessions
* Charging provisioning for WebRTC sessions
* Collection of consumption and QoE metrics data related to WebRTC sessions
* Offering ICE functionality such as STUN and TURN servers
* Offering WebRTC signaling servers, potentially with interoperability to other signaling servers

The provisioning interface is not relevant to all collaboration scenarios and some of the 5G support functionality may be offered without application provider provisioning.

The RTC-5 interface is an interface between the Media Session Handler (MSH) and the 5G-RTC AF. It is used to convey configuration information from the AF to the MSH and to request support for a starting/ongoing WebRTC session. The configuration information may consist of static information such as the following:

* Recommendations for media configurations
* Configurations of STUN and TURN server locations
* Configuration about consumption and QoE reporting
* Discovery information for WebRTC signaling and data channel servers and their capabilities

The support functionality includes the following:

* MSH informs the MSH about a WebRTC session and its state
* MSH requests QoS allocation for a starting or modified session
* MSH receives notification about changes to the QoS allocation for the ongoing WebRTC session
* MSH receives updates exchanges information about the WebRTC session with the 5G-RTC STUN/TURN/Signaling Server, e.g. to identify a WebRTC session and associate it with a QoS template

The 5G-RTC functionality that offer application functions to the WebRTC application (blocks in yellow) can equally be provided by Application Servers (5G-RTC AS) instead of AFs. These could then use a dedicated interface RTC-3 to request configurations and network support for the ongoing WebRTC sessions from the 5G-RTC AF.

The MSH is a function in the UE that provides access to 5G-RTC support functions to the WebRTC applications. These functions may be offered on request, i.e. through the RTC-6 interface, or transparently without direct involvement of the application. The MSH may for instance assist indirectly in the ICE negotiation by providing a list of STUN and TURN server candidates that offer 5G-RTC functionality. The MSH also collects QoE metric reports and submits consumption reports. They may also offer media configuration recommendations to the application through RTC-6. The RTC-6 interface may be offered to both native and web-based WebRTC applications equally. The interface should be defined in an abstract form, e.g. using IDL or WebIDL, which would allow implementers to map it to their desired implementations and environments (e.g. make them accessible through a JavaScript API for web-based applications). RTC-6 is completely orthogonal to the WebRTC APIs developed by W3C and should offer the following functionality:

* allows the application to indicate that a new session is starting or stopping
* allows the application to retrieve the MNO configuration for WebRTC session, such as URLs for trusted ICE functions, the recommended QoS template, the QoE reporting requirements, etc.
* provides the application with recommendations on the media and QoS configuration

The RTC-4 interface applies to some RTC functions that assure media relaying and processing functionality for the WebRTC session. It represents the user plane data exchange to and from the UE and is fully compatible with WebRTC.

Finally, RTC-8 will be used to exchange information between the application and the Application Provider.

# Proposal

We propose to agree the baseline architecture and the interface functionality in clause 2 and document them in the iRTCW PD.