**Agenda item:** 8.7

**Source:** Qualcomm Inc., BBC, Tencent

**Title: EMSA Architecture**

**Document for** Discussion andAgreement

# Introduction

In this contribution, we propose a solution for the mapping of the 5G edge architecture as developed by SA6 and the management architecture for edge as developed by SA5 onto the 5GMS architecture. This is a follow-up on an earlier contribution that offered 2 options for the mapping. The goal is to avoid the definition of any new interfaces between the 5GMS elements and the other architecture elements.

# EMSA Architecture

The EMSA architecture is an integration of the 5GMS architecture, the SA6 Edge architecture and the SA5 management architecture. The EMSA architecture is depicted in the following figure:



In this approach, the EEC, EES, and EAS are implemented as part of the MSH, 5GMS AF, and 5GMS AS, respectively. This approach does not exclude the standalone existence of these functions, it merely implies that for the media vertical, these functions are implemented as part of 5GMS architecture functions.

This also implies the following:

* A Media Session Handler (MSH) that is edge-enabled is required to implement the EDGE-5 API, which could be part of the M6 interface.
* A 5GMS AF that is edge-enabled is required to realise EES functionality
* A 5GMS AF that is edge-enabled is required to expose the EDGE-3 interface towards the EAS function of 5GMS AS instances.
* A 5GMS AF that is edge-enabled is required to register with an ECS function using the EDGE-6 interface.
* A 5GMS AF that is edge-enabled may be required to implement the EDGE-9 interface to support media session relocation.
* A 5GMS AF that is edge-enabled is required to implement the EDGE-1 interface to support registration and provisioning of EEC functions, and discovery by them of EAS instances.
* A 5GMS AF may perform compute resource allocation using the MnS-C interface.

This architecture supports both client-driven as well as AP-driven management of the edge processing session.

In the client-driven approach, the application is aware of the edge processing and takes steps, such as using the EDGE-5 APIs to discover and locate an application server in the edge DN.

In AP-driven approach, the UE application may discover and connect to the closest EAS without significant support from the EEC and potentially also without the application being aware of the edge processing. The 5GMS AF acts on behalf of the 5GMS Application Provider to allocate processing resources based on the application needs. In order for this transparent resolution to work, SA2 is expected to define a solution to allow the AF to influence the DNS resolution for the domain name(s) related to the media distribution by the Application Provider as described in clause X.

# Generic Call Flow for Session Establishment

A detailed call flow for the session establishment based on the EMSA architecture is as follows:



1. Edge Computing Provisioning
	1. Spawn ECS
	2. Spawn 5GMS AF
	3. EES Configuration
	4. EES Registration with ECS
2. 5GMS Application Provider Provisioning
3. Create Provisioning Session
4. Provision 5GMS features
5. UE Edge Computing Discovery
6. Application Initialization
7. Locate EAS/5GMS AS
8. Locate local EES
9. Locate EAS/5GMS AS
10. Check resource template
11. Instantiate new EAS/5MGS AS
12. Spaw 5GMS AS instance
13. EAS configuration
14. Register EAS with EES
15. Configure provisioned features
16. 5GMS AS location response
17. 5GMS AS location response
18. 5GMS Session
19. Start session
20. Session starting event
21. Retrieve service access information
22. Media transfer
23. Method calls and notifications
24. Reporting, network assistance, and dynamic policy
25. End session
26. Session ending event
27. Session stop and final reporting

# Proposal

We propose to agree the proposed architecture mapping and to add the content of section 2 to the EMSA TR 26.803.