Network Capability Exposure through APIs - Resources and Outcomes

- Check the **Execution Plan**

Content Production and Contribution Scenarios

Wireless connectivity is a key element in content production and contribution, including productions in studios, coverage of live events in venues or on-the-move, commentary boxes, newsgathering,... These different setups may have unique needs involving specific infrastructure and equipment, as well as distinct media connectivity flows with varying quality-of-service (QoS) requirements. To enhance the standard best-effort connectivity, Network APIs can be used to access advanced network capabilities. The following analysis includes a brief description of scenarios where advanced network capabilities can be invoked, example operational workflows and a series of requirements (including network capabilities) in the context of content production and contribution.

- Scenarios: Network Capability Exposure and APIs for Content Production and Contribution.
- Workflows: Required interactions to exploit network capabilities.
- Requirements: Network Capabilities (Network Services) for Content Production & Contribution.

Live Media Streaming

- Scenarios: Network Capability Exposure and APIs for Live Media Distribution.
- Workflows: Required interactions to exploit network capabilities.
- Requirements: Network Capabilities (Network Services) for Content Production & Contribution.

Network APIs: Industry Initiatives

- Network API Initiatives under analysis including CAMARA Project and 3GPP APIs

TABLE OF CONTENTS

WARNING

This documentation is currently **under development and subject to change**. It reflects outcomes elaborated by 5G-MAG members. If you are interested in becoming a member of the 5G-MAG and actively participating in shaping this work, please contact the Project Office

Network Capability Exposure and APIs for Content Production and Contribution

Wireless connectivity plays a key role in content production and contribution, with scenarios such as production in studios, coverage of live events in venues or on-the-move, commentary boxes, newsgathering,... These different setups may have unique needs involving specific infrastructure and equipment, as well as distinct media connectivity flows with varying quality-of-service (QoS) requirements.

Media connectivity flows and QoS

Examples of media flows include: Content acquisition, contribution to remote locations, control signals and telemetry, performance and analytics, voice services, return video/audio,...

Each of them may involve certain QoS requirements including: data rate, bit error rate, latency, jitter,... Radio-related parameters such as amount of spectrum, uplink/downlink rates, or resilience to interference are considered as tunable variables to meet the QoS requirements and therefore are not considered up-front requirements.

Meeting such requirements may involve a certain degree of control on the network and/or the network setup. While in some cases it may be possible to meet the requirements via best-effort contended public networks, others may require the setup of non-public networks (NPN) and invoking certain network capabilities beyond best-effort connectivity.

Exposure of network capabilities

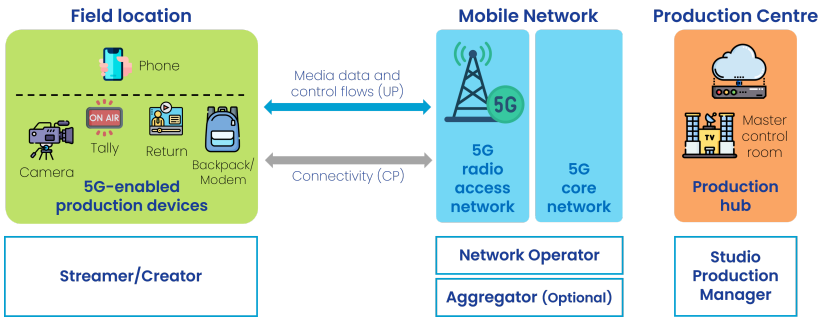
The exposure of network capabilities to applications representes an opportunity to exploit advanced network features beyond best-effort connectivity. Examples of network capabilities maz include on-demand quality, user equipment (UE) management, precise time synchronization,... Accessing and utilizing the desired features can be intricate and inconsistent across different networks. Several initiatives are taking shape to explore the opportunities behind Network APIs (exposing network capabilities to API consumers), offering high-level abstractions of underlying network functionalities to simplify resource utilization for non-network experts.

Example: Media connectivity flows for newsgathering and mobile journalism

Description

A journalist in the field or at a venue is capturing and contributing (uplinking) content to an application server located in the cloud or remote premises. The content is delivered using one of the following options:

- A single UE (e.g. a smartphone) equipped with a single SIM card (or eSIM) connected to the mobile network.
- A single device (e.g. a smartphone) equipped with 2 UEs each with 1 SIM card (or eSIM) connected to a different carrier of the same mobile network or different mobile networks. Note that multi-SIM devices enable users to utilize multiple cellular connections simultaneously. Dual-SIM Dual-Active (DSDA) enable this use case with two SIM cards. This is different to Dual-SIM Dual-Standby (DSDS), which allows only one SIM to stay connected with active data at a time. DSDA enhances data performance for end users by enabling the use of two data connections concurrently across SIM1 and SIM2, with the option to choose the best of them or aggregate both, if necessary, to reach higher data throughput.
- A device with multiple UEs (e.g. a cellular bonding backpack) equipment with multiple SIM cards each one connected to a different carrier of the same mobile network or connected to different mobile networks.



Actors involved

The actors involved are:

- **Streamer/Creator**, uses the content acquisition equipment to capture media, uses the network and sends data to the server.
- a **Studio Production Manager**, located e.g. within the production centre.
- **Network Operator**, provides the network used for the production. A set of network capabilities can be configured through APIs (referred to as Network APIs in the following).
- **Aggregator (optional)**, provides access to the network capabilties of different Network Operators. See GSMA Open Gateway and GSMA Operator Platform as examples.

Network functions and applications involved

- **Production Device** (such as a smartphone or a camera attached which is connected to a modem or a backpack), used by the streamer/creator. The device contains at least one UE with a Subscription (SIM) and can host one or more client applications. A client application can be a video capturing and encoding application, which generates and sends a continuous video stream to a receiving Media Server.
- **Network API Platform**, used by the Network Operator for exposing network Capabilities. The Network API Platform offers a collection of functions e.g. for Authentication / Authorization of the API Invoker (AuthZ Function) and different API Provider functions for different network capabilities. Beside this, there may be more functions, e.g. for API usage metering, API usage throttling, etc.
- **Aggregator API Platform (optional)**, located in the path between the Network API Platforms and the API Invoker. It grants access to Network API Platforms from different Network Providers.
- **API Consumer / Invoker**, used by the Production equipment (functions) to interact with the Network API Platform of a Network Provider.
- **Media Servers**, typically located in the Studio Production Hub (operated by the Production Manager) and interact with the production devices, e.g. receiving video or audio streams.

Example: Media connectivity flows for Outside Broadcast

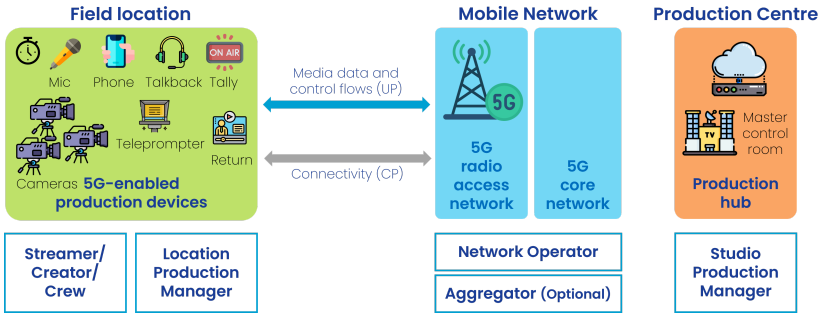
Description

A production crew deployed in the field or at a venue is covering an event using multiple devices including cameras, audio equipment, intercom, etc. Multiple devices are concurrently used during the production. Not all data flows have the same priority and quality requirements. Therefore, each device and data flow should get the requested connectivity performance (e.g. throughput, latency, jitter,...) and with the desired QoS, which may change for each device and data flow during the production. Two options are considered when it comes to network deployment:

- A network deployed in the field or at a venue that is used to connect devices and manage the production locally. Devices capturing and contributing (uplinking) content deliver it to an

application server located in the cloud or in the field location. The final program output may be generated locally and delivered to the production centre using one of the options described in the “newsgathering and mobile journalism” scenario, for instance, by means of a device connected to a mobile network. The scenario may involve the deloyment of different networks:

  - An SNPN, deployed locally. Remote connectivity can be provided by means of a fiber connection or a public network (PNI-NPN) to which the SNPN is connected to. However, the networks are detached and traffic from devices is not directly contributed but only the program output.
  - A PNI-NPN which provides dedicated connectivity locally and for the production devices.
- A public network to which devices are connected to. Devices are managed remotely and are contributing data to the production centre. In this case, a PNI-NPN may be used to guarantee QoS for the different flows carried across the mobile network.



Actors involved

The actors involved are:

- A **Production Manager**, deals with the configuration of the production equipment and the access network and has the authority to use the application that interacts with the network operator. It is either:
  - a Location Production Manager, who is together with the Production Crew in the field, or
  - a Studio Production Manager, who is located e.g. within the production centre.
- **Streamer/Creator/Crew**, uses the content acquisition equipment to capture media and the network to send data to the server.
- **Network Operator**, provides the network used for the production. A set of network capabilities can be configured through APIs (referred to as Network APIs in the following).
- **Aggregator (optional)**, provides access to the network capabilties of different Network Operators. See GSMA Open Gateway and GSMA Operator Platform as examples.

Network functions and applications involved

- **Production Devices** (such as a Camera), used by the crew during a production. Each device contains at least one UE with a Subscription (SIM) and can host one or more client

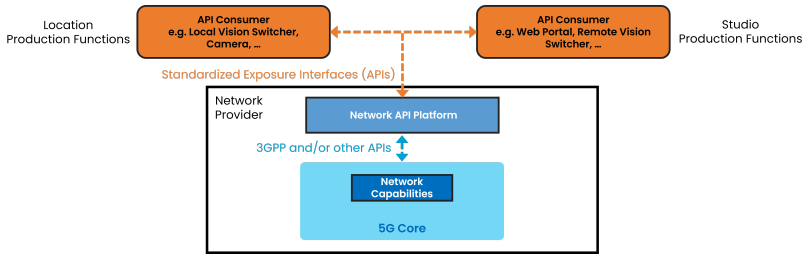
applications. A client application can be a video capturing and encoding application, which generates and sends a continuous video stream to a receiving Media Server.

- **Network API Platform**, used by the Network Operator for exposing network Capabilities. The Network API Platform offers a collection of functions e.g. for Authentication / Authorization of the API Invoker (AuthZ Function) and different API Provider functions for different network capabilities. Beside this, there may be more functions, e.g. for API usage metering, API usage throttling, etc.
- **Aggregator API Platform (optional)**, located in the path between the Network API Platforms and the API Invoker. It grants access to Network API Platforms from different Network Providers.
- **API Consumer / Invoker**, used by the Production equipment (functions) to interact with the Network API Platform of a Network Provider.
- **Media Servers**, typically located in the Studio Production Hub (operated by the Production Manager) and interact with the production devices, e.g. receiving video or audio streams.

Collaboration scenarios for the provisioning of network capabilities

Collaboration scenario #1: Direct invoking Network APIs

The **Network API Platform** of a Network Operator is accessed directly from **API Consumers**, either deployed with the Studio Production or the Location Production functions. The API consumer can be a Web Portal, e.g. offered by the CSP. Alternatively, the API consumers can be embedded production devices like a Vision Mixer or a production orchestration solution (NMOS concept). The API consumer functions can be integrated in media servers, responsible for receiving the video stream from the application client.



Collaboration scenario #2: Invoking Network APIs via an Aggregator

The **Network API Platform** of a Network Operator is accessed via an **Aggregator API Platform**. The Aggregator Platforms harmonize capabilities offered by different Network Providers and routes customer requests to them.

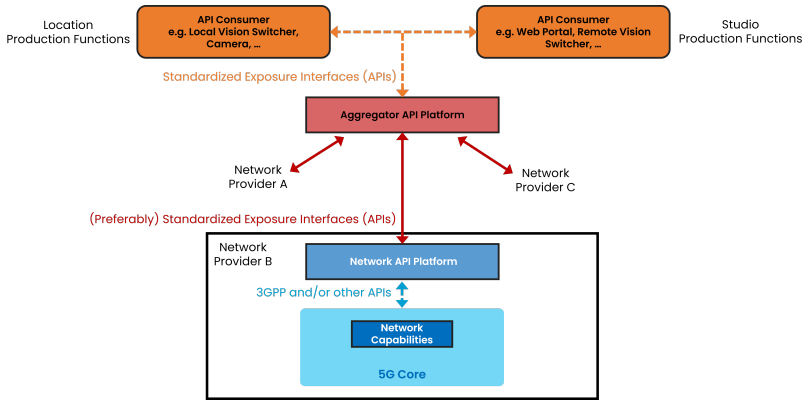


TABLE OF CONTENTS

WARNING

This documentation is currently **under development and subject to change**. It reflects outcomes elaborated by 5G-MAG members. If you are interested in becoming a member of the 5G-MAG and actively participating in shaping this work, please contact the Project Office

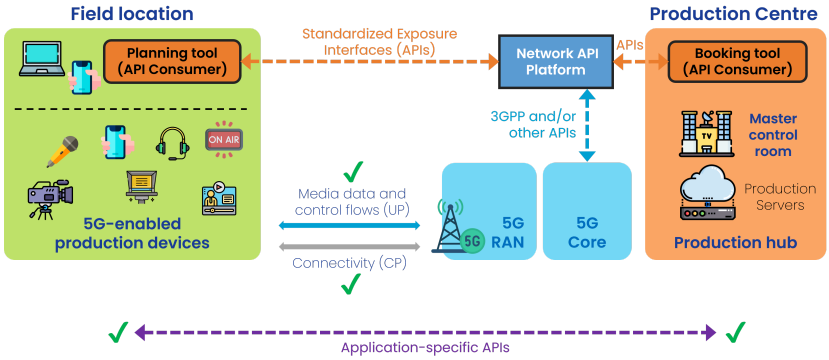
Required interactions to exploit network capabilities

Pre-conditions

- The production company has set up an agreement with a network operator for usage of certain **network capabilitues** (e.g. selected from an API catalogue) and has received authentication credentials from the newotk operator authorising their use (when available).
- The production crew (on location or located in the production centre) has access to one or several **Network API Platforms**. These platforms are accessible through any device/connectivity (e.g. Internet-acccessible website portal, command line tools, dedicated application, etc.).
  - Note: For Network API Platform access, the production crew has obtained key access tokens/keys/credentials/payment details in advance.
- The production crew has a set of credentials (SIM/eSIM) for the network the production device nodes will connect to.
- By default, the network provides “best efforts” connectivity.
- Production device nodes already have working “best efforts” connectivity to the network.

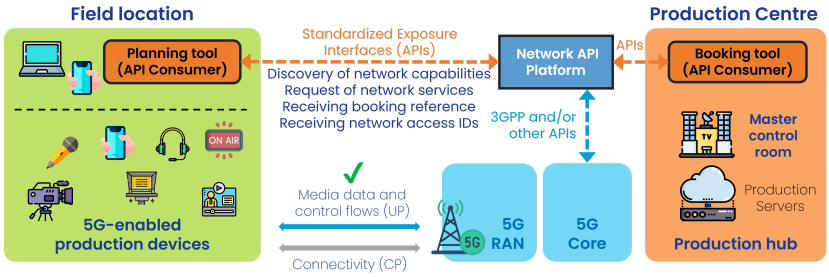
Before the Event

Phase A: Preparing devices, configuring application clients and servers, and configuring client/server flows



- Some production device nodes are UEs; others are connected to the Data Network:
  - Example production device nodes connected to the RAN: wireless cameras, wireless camaera control units, wireless microphones, wireless talkback intercom, etc.
  - Example production device nodes connected to the Data Network: vision mixer, sound mixer, etc.
- An **application-specific API** enables communication between the production network orchestrator and the production device nodes.

Phase B: Event planning and pre-booking



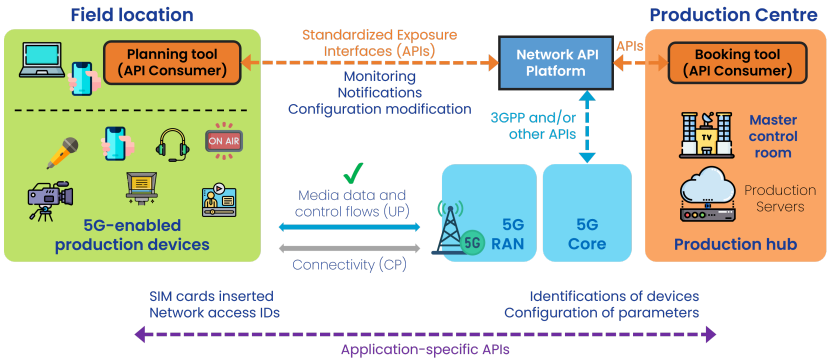
1. Through the Network API Platform, the production crew (on location or located in the production centre) can discover the capabilities the network can offer in a particular location and at a particular time (for which the production company is eligible for).
   - Example: QoD available, connectivity monitoring available, Timing as a service available, edge compute instantiation, etc.
2. Through the Network API Platform, the production crew requests network services for each of the planned SIM cards in advance. Possible services (network capabilities) are:
   a. *Quality-on-Demand*
      - One or several QoS profiles for each SIM card (QoS profiles are mapped to 5QIs)
      - Example: A sim may be pre booked for One uplink video / One uplink audio / one downlink data / etc.
   b. *Time-as-a-service*
      - Provided either by access stratum or Precision Time Protocol (PTP).

Note: Booking is done based on:

- Geographical area
- Schedule (starting time and closing time of the event)

1. Through the Network API Platform the production manager receives a booking reference responding to the service request.
2. Through the Network API Platform the production manager accepts the service booking offer (involving payment/contract/SLA aspects).
3. Through the Network API Platform the production manager receives **network access IDs** to be used by the production device UEs to access the network on location.
    - Each network access ID ultimately resolves to a Data Network Name (DNN) and optionally a network slice identifier (S-NSSAI).
4. The production crew is responsible for inserting the right SIM card into the right production device UE.
    - Additional configuration of the network access ID may also be required (e.g. by using OMA-DM).

During the event



Phase C: Location setup and configuration

1. Production crew arrives in the venue, plugs the SIM cards and turn on the devices, connectivity is enabled based on the booked network services (See phase B).
2. The production crew initiates the setup of the location production by interacting with the production network orchestrator.
3. The production network orchestrator configures the production device nodes using an application-specific API, citing the network access IDs delivered in step B.5).
    - Example: QoD service: A camera for which one video + one audio is pre-booked. The application-specific API is used to properly configure the bitrate of the audio and video output, and the provided IDs.
    - Example: Time Sync service: A camera for which access to global clock is requested. The application-specific API is used to properly configure the time parameters and the

provided IDs.

Independent steps that can be triggered during the event

- The production crew can use the Network API Platform to monitor that the flows are coming and are properly using the reserved resource.
- The production crew receives notification through the Network API Platform indicating potential issues (throughput, delay, etc.).
- The production crew through the Network API Platform can request a change of the current configuration.
- Same validation steps as from B.2 to B.5 will be conducted after requesting the change.
- Changes can be, for example:
  - Switch profile A from SIM card A to SIM card B.
  - Increase or decrease the capacity of an existing profile.
  - Remove or add a profile to a SIM card.
  - Enable/Disable time service on a SIM card.
  - etc.

Note: Network access IDs are not expected to change when a reconfiguration occurs.

Note: the steps in phase C are repeated whenever a service is added and created from scratch.

After the event

Phase D: Location teardown

1 Through the Network API Platform, the production crew releases the booked resources.

WARNING

This documentation is currently **under development and subject to change**. It reflects outcomes elaborated by 5G-MAG members. If you are interested in becoming a member of the 5G-MAG and actively participating in shaping this work, please contact the Project Office

Network Capabilities (Network Services) for Content Production & Contribution

This is a list of network capabilities required to realize the scenarios described in Network Capability Exposure and APIs for Content Production and Contribution Scenarios.

Media delivery with Quality of Service (QoS)

Requirement	API
Ability to request different QoS profiles for individual data flows coming from the same production device node	
Ability to separate media/data flows coming from the same production device node	
Delivery to endpoint (Application Media Server) may be identified by security/protocol/IP/port	
Ability to configure new or re-configure existing QoS profiles to be selected during runtime	
Ability to select at runtime a QoS profile for a media flow	
Ability to receive ACK (success/fail)	

Information monitoring, logging and/or Network assistance

Requirement	API
Ability to receive information from the network	
Real-time information for QoS profile re-selection and/or e.g. codec reconfiguration, bitrate reconfiguration	
Information during runtime for troubleshooting	
Information after the session (logging information) for post-processing	

Time Synchronization

Requirement	API
Ability to enable distribution of timing information	

Voice service for Intercom

Requirement	API
Ability to establish a voice service across the intercom devices deployed at the production location or between the production center and the production location	

> **NOTE**
> Focus on the QoS for Intercom - a voice service offered by the network may not be so relevant (alternative solutions, WebRTC). But multicast, MCPTT may be of use.

Considerations on Devices

Identification of devices

Requirement | API – | – Devices should be uniquely identifiable during operation | Devices should be dynamically added or deleted during operation and attachable to given network capabilities | Each device should only access the network capabilities which have been assigned during booking |

Device on-boarding and API consumer on-boarding

Requirement | API – | – TBD How to obtain credentials |

Discovery of network capabilities

Requirement | API – | – TBD How to discover network capabilities |

WARNING

This documentation is currently **under development and subject to change**. It reflects outcomes elaborated by 5G-MAG members. If you are interested in becoming a member of the 5G-MAG and actively participating in shaping this work, please contact the Project Office

Network API Initiatives under analysis

3GPP APIs for Quality of Service

We collect here information about the following 3GPP APIs

NEF

- Nnef_AFSessionWithQoS
- Nnef_ChargeableParty
- Nnef_BDTPNegotiation

PCF

- Npcf_PolicyAuthorization
- Npcf_BDTPolicyControl

CAMARA APIs for Quality of Service

Relevant APIs

- Quality On Demand APIs
- QoS Booking APIs
- Dedicated Networks APIs
- Network Slice Booking APIs
- Connectivity Insights APIs

Relevant QoS Parameters

QUALITY ON DEMAND

Name	Description
targetMinUpstreamRate	This is the target minimum upload speed for the QoS profile. It represents the minimum rate that the network attempts to deliver. Please note that this is a target value—the network might not always be able to provide this rate under all conditions. It helps ensure that applications like video calls or live streaming perform consistently.
maxUpstreamRate	The maximum best effort data
maxUpstreamBurstRate	When defined, this is the maximum upstream burst rate for the QoS profile, that will enable the network to burst data at a higher rate than the maxUpstreamRate for a period of time.
targetMinDownstreamRate	This is the target maximum upload speed for the QoS profile. It represents the maximum rate that the network attempts to deliver. Please note that this is a target value—the network might not always be able to provide this rate under all conditions. It helps ensure that applications like video calls or live streaming perform consistently.
maxDownstreamRate	The maximum best effort rate
maxDownstreamBurstRate	When defined, this is the maximum downstream burst rate for the QoS profile, that will enable the network to burst data at a higher rate than the maxDownstreamRate for a period of time. This can result in improved user experience when there is additional network capacity. For instance, when a user is streaming a video, the network can burst data at a higher rate to fill the buffer, and then return to the maxUpstreamRate once the buffer is full.
minDuration	The shortest time period that this profile can be deployed.
maxDuration	The maximum time period that this profile can be deployed. Overall session duration must not exceed this value. This includes the initial requested duration plus any extensions.
priority	Priority levels allow efficient resource allocation and ensure optimal performance for various services in each technology, with the highest priority traffic receiving preferential treatment. The lower value the higher priority. Not all access networks use the same priority range, so this priority will be scaled to the access network’s priority range.
packetDelayBudget	The packet delay budget is the maximum allowable one-way latency between the customer’s device and the gateway from the operator’s network to other networks. By limiting the delay, the network can provide an acceptable level of performance for various services, such as voice calls, video streaming, and data. The end-to-end or round trip

Name	Description
	latency will be about two times this value plus the latency not controlled by the operator
jitter	The jitter requirement aims to limit the maximum variation in round-trip packet delay for the 99th percentile of traffic, following ITU Y.1540 standards. It considers only acknowledged packets in a session, which are packets that receive a confirmation of receipt from the recipient (e.g., using TCP). This requirement helps maintain consistent latency, essential for real-time applications such as VoIP, video calls, and gaming.
packetErrorLossRate	This field specifies the acceptable level of data loss during transmission. The value is an exponent of 10, so a value of 3 means that up to 10^{-3}, or 0.1%, of the data packets may be lost. This setting is part of a broader system that categorizes different types of network traffic (like phone calls, video streams, or data transfers) to ensure they perform reliably on the network.

QUALITY ON DEMAND - EXPERIMENTAL PARAMETERS - L4S

NOTE: l4sQueueType is experimental and could change or be removed in a future release. Specifies the type of queue for L4S (Low Latency, Low Loss, Scalable Throughput) traffic management. L4S is an advanced queue management approach designed to provide ultra-low latency and high throughput for internet traffic, particularly beneficial for interactive applications such as gaming, video conferencing, and virtual reality. For more details, refer to the L4S standard

NOTE: serviceClass is experimental and could change or be removed in a future release. The name of a Service Class, representing a QoS Profile designed to provide optimized behavior for a specific application type. While DSCP values are commonly associated with Service Classes, their use may vary across network segments and may not be applied throughout the entire end-to-end QoS session. This aligns with the serviceClass concept used in HomeDevicesQoQ for consistent terminology. Service classes define specific QoS behaviors that map to DSCP (Differentiated Services Code Point) values or Microsoft QoS traffic types. The supported mappings are:

- 1 Values aligned with the RFC4594 guidelines for differentiated traffic classes.
- 1 Microsoft QOS_TRAFFIC_TYPE values for Windows developers.

QOS BOOKING APIS

Makes use of the QoS Profiles retrieved by the QoS Profiles API (part of Quality on Demand APIs).

NETWORK SLICE BOOKING

Name	Description
MaxNumofTerminals	
DLThroughputPerTerminal	
ULThroughputPerTerminal	
DLLatency	DLLatency is an attribute specifies the required DL packet transmission latency (millisecond) through the 5G network.
ULLatency	ULLatency is an attribute specifies the required UL packet transmission latency (millisecond) through the 5G network.

DEDICATED NETWORKS

Name	Description
maxNumberOfDevices	
aggregatedUlThroughput	
aggregatedDlThroughput	
DLLatency	DLLatency is an attribute specifies the required DL packet transmission latency (millisecond) through the 5G network.
ULLatency	ULLatency is an attribute specifies the required UL packet transmission latency (millisecond) through the 5G network.

CONNECTIVITY INSIGHTS

Name	Description
packetDelayBudget	the maximum allowable one-way latency between the customer's device and the gateway from the operator's network to other networks. The end-to-end or round trip latency will be about two times this value plus the latency not controlled by the operator
targetMinDownstreamRate	This is the target minimum downstream rate.
targetMinUpstreamRate	This is the target minimum upstream rate.
packetlossErrorRate	The exponential power of the allowable error loss rate 10^(-N). For 5G network the 3GPP specification TS 23.203 defines the packet error loss rate QCI attribute.

Name	Description
jitter	Aims to limit the maximum variation in round-trip packet delay for the 99th percentile of traffic, following ITU Y.1540 standards. It considers only acknowledged packets in a session, which are packets that receive a confirmation of receipt from the recipient (e.g., using TCP).
signalStrength	rough indication of the end user device radio signal conditions
connectivityType	the access technology connecting the user device to the operator network

Information

- CAMARA Repositories in GitHub: Link
- API Backlog: Link
- Proposed new APIs: Link