Southbound Interface Network Resources APIs

Version 2.0

29 March 2023

This is a Non-binding Permanent Reference Document of the GSMA

Security Classification: Non-confidential

Copyright Notice

Disclaimer

Compliance Notice

The information contain herein is in full compliance with the GSM Association's antitrust compliance policy.

This Permanent Reference Document is classified by GSMA as an Industry Specification, as such it has been developed and is maintained by GSMA in accordance with the provisions set out in GSMA AA.35 - Procedures for Industry Specifications.

Table of Contents

1 Introduction 4
1.1 Overview 4
1.2 Scope 4
1.3 Definitions 5
1.4 Abbreviations 5
1.5 References 6
1.6 Conventions 6
1.7 Summary SDO Reference Mapping Table 7
2 Network Integration Support APIs in SBI-NR Interface 7
2.1 Network Events API 7
2.1.1 Description 7
2.1.2 Requirements and Service Aspects 7
2.1.3 Procedures 8
2.1.4 API 8
2.2 QoS Management API 8
2.2.1 Description 8
2.2.2 Requirement and Service Aspects 9
2.2.3 Procedures 9
2.2.4 API 9
2.3 Traffic Influence API 9
2.3.1 Description 9
2.3.2 Requirement and Service Aspects 10
2.3.3 Procedures 10
2.3.4 API 10
2.4 Application Relocation API 10
2.4.1 Description 10
2.4.2 Requirement and Service Aspects 10
2.4.3 Procedures 11
2.4.4 API 11
2.5 Confirm User Location API 11
2.5.1 Description 11
2.5.2 Requirement and Service Aspect 11
2.5.3 Procedures 12
2.5.4 API 12
2.6 Mobility Triggers API 12
2.6.1 Description 12
2.6.2 Requirement and Service Aspects 12
2.6.3 Procedures 12
2.6.4 API 13
2.7 Mobility Control API 13
2.7.1 Description 13
2.7.2 Requirements and Service Aspects 13
2.7.3 Procedures 13

2.7.4 API 14
3 Network key enabling capabilities 14
3.1 End user Public IP address to MSISDN mapping. 14
3.1.1 Introduction 14
3.1.2 Network setup 15
3.2 User Info API 16
3.2.1 GET Method: IdentifyUser 16
3.2.2 Data Model 17
Annex A Open API definitions 17
A.1 User Info API 17
Annex B Document Management 25
B.1 Document History 25
B.2 Other Information 25

1 Introduction

1.1 Overview

The Operator Platform's (OP) Network Integration Support Application Programming Interfaces (APIs) are mainly related to the Telco network interfaces, e.g. the Southbound Interface-Network Resources (SBI-NR), but also aiming to provide the end-to-end fulfilment of the specific topic, hence including any other OP Interface as described in GSMA PRD OPG.02 [1].

The list of the topics and the associated APIs in the SBI-NR interface are presented in the Table 1 below. Grey items are still under analysis and will be available in a later release.

No	OPAG APIs in SBI-NR Interface (Phase 1)	OP Interface
1	Network Events	SBI-NR
2	QoS Management	SBI-NR
3	Traffic Influence	SNI-NR
4	Application Relocation	SBI-NR
5	Confirm User Location	SBI-NR
6	Mobility Triggers	SBI-NR
7	Mobility Control	SBI-NR
8	Managing Service Availability in LADN	SBI-NR
9	User Location Privacy Indicator	SBI-NR
10	User Authentication and Authorisation (defined in UNI)	SBI-NR

Table 1: List of APIs in SBI-NR Interface

The purpose of this document is to provide the API requirements and the Standards Developing Organisation (SDO) reference mapping for each API listed.

The structure to document each API consists of:

1. Descriptions: summary of the purpose and expected use of the API
2. Requirements and Service Aspects: References to requirements related to the API identified in GSMA PRD OPG.02 [1].
3. Procedures: References to procedures and flows from an SDO's specifications (e.g. 3GPP) that match with the OP's view for the API.
4. API: References to API (API parameters, HTTP implementation and YAML file) located in SDOs specifications.

The main reference source of the API Requirements is the GSMA PRD OPG.02 [1]. Other sources for the API Requirements are found in the relevant SDO's reference documents definitions.

1.2 Scope

The present document aims to define OP APIs that are related to SBI-NR Interface in the OP architecture. This set of APIs define the Network Integration Support APIs.

1.3 Definitions

Term	Description
Network Events	The OP in the role of Application Function (AF) would need to manage network events and notifications over the SBI-NR interface (NEF APIs) and orchestrate edge Application Instances in target Cloudlets and synchronise the associated application states to provide application Session Continuity. [1]
Application Relocation	The Operator Platform shall be able to consider the application-specific requirements for managing mobility over different edge nodes. [1]
Mobility Triggers	Many different elements shall monitor and control the end-to-end service delivery for detecting any modification and trigger a change on the path. [1]
Mobility Control	Because of this User Equipment (UE) mobility, or because of the OP's measurements or knowledge, or hints from the application about performance degradations, the OP may decide that a different edge compute resource can better host the Edge Application. An OP may be able to negotiate the UE data plane mobility process based on the application instance relocation process. An OP may be able to specify the request for routing, influencing network mobility and routing. [1]
Traffic Influence	Based on some of the events on the SBI-NR interface, e.g. location monitoring events, QoS status notification events etc., the OP may determine the level of QoS provided by the mobile network to application sessions against the Quality of Service (QoS) level requested by the application. In such cases, the OP may initiate the user plane relocation (e.g., by using Traffic Influence APIs) services on the SBI-NR interface. Possibly this may result in the triggering of session mobility procedures in the mobile network. [1]

1.4 Abbreviations

Term	Description
5QI	5G QoS Identifiers
AF	Application Function
API	Application Programming Interface
AS	Application Server
CGNAT	Carrier Grade NAT
EAC	Edge Application Client
EAS	Edge Application Server
EEC	Edge Enabled Client
EES	Edge Enabled Server
EWBI	East West Bound Interface
DDN	Downlink Data Notification
HTTP	Hypertext Transfer Protocol
IMEI	International Mobile Equipment Identity
IMEI(SV)	International Mobile Equipment Identity (Software Version)
IMSI	International Mobile Subscriber Identity
IP	Internet Protocol
MSISDN	Mobile Subscriber Integrated Services Digital Network Number

Term	Description
NAT	Network Address Translation
NBI	Northbound Interface
NEF	Network Exposure Function
OP	Operator Platform
OPG	Operator Platform Group
PDN	Packet Data Network
PRD	Permanent Reference Document
QoE	Quality of Experience
QoS	Quality of Service
SBI-NR	Southbound Interface – Network Resources
SCEF	Service Capability Exposure Function
SDO	Standards Developing Organisation
UDM	Unified Data Management
UE	User Equipment
UNI	User Network Interface
YAML	YAML Ain't Markup Language. YAML is *a human-readable data serialization standard* that can be used in conjunction with all programming languages
V2X	Vehicle to X (anything)

1.5 References

Ref	Doc Number	Title
[1]	OP PRD	GSMA PRD OPG.02, 14 April 2022
[2]	RFC 2119	"Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997. Available at http://www.ietf.org/rfc/rfc2119.txt
[3]	3GPP TS 29.522	5GS, Network Exposure Function Northbound APIs, Stage 3, Release 17.6, June 2022
[4]	3GPP TS 23.558	5GS, Architecture for enabling Edge Applications, Release 17, June 2022
[5]	3GPP TS 29.558	5GS, Enabling Edge Applications API Specifications, Stage 3, Release 17, June 2022
[6]	3GPP TS 29.591	5GS, Network Exposure Function Southbound Services, Stage 3, Release 17, June 2022
[7]	3GPP TS 23.502	Procedures for the 5G Systems (5GS), Stage 2, Release 17, June 2022

1.6 Conventions

The key words "must", "must not", "required", "shall", "shall not", "should", "should not", "recommended", "may", and "optional" in this document are to be interpreted as described in RFC2119 [2].

1.7 Summary SDO Reference Mapping Table

The OP's SBI-NR Interface is linking with the 3GPP 5G Core Network Exposure Function (NEF). Table below summarises the SDO Reference mapping between the OP SBI-NR and 3GPP NEF in its Northbound Interface (NBI) and associated APIs.

No	APIs in SBI-NR Interface	OP Interface	SDO Reference Mapping - 3GPP
1	Network Events	SBI-NR	3GPP TS 29.522
2	QoS Management	SBI-NR	3GPP TS 29.522/29.122
3	Traffic Influence	SNI-NR	3GPP TS 29 522
4	Application Relocation	SBI-NR	3GPP TS 29 558
5	Confirm User Location	SBI-NR	3GPP TS 29.522/29.122 3GPP TS 29.558
6	Mobility Triggers	SBI-NR	3GPP TS 29.522/29.122
7	Mobility Control	SBI-NR	3GPP TS 29.522

2 Network Integration Support APIs in SBI-NR Interface

2.1 Network Events API

2.1.1 Description

Goal of the OP's Network Events service is to provide interface for Operators and Application Providers to support enhanced application-aware exposure for 5G network events including federated topologies. Two interfaces between 5G System, Edge System and Application Client should be described:

- **Collecting Network Status Events SBI-NR** which supports User Plane related notifications (Session Establish, UE IP address allocation, Access and Mobility with Registration, Connection, Reachability and Mobility Management) and other events related to application traffic. Those events are mostly exposed by the 4G Service Capability Exposure Function (SCEF)/5G NEF.
- **Network Events NBI** exposed by an OP to an Application Provider allowing introducing context awareness regarding occurred network events and provide support for quality, performance and security to ensure the required Quality of Experience (QoE) of the OP's customers.

2.1.2 Requirements and Service Aspects

The requirements for the API are specified in the GSMA PRD OPG.02 [1]. in the following sections:

Section	Document	Reference Section Title
3.5.1.7.3	OPG.02	Network Event Support (*User Equipment (UE) location information and events; * UE network connection events; *Application to UE connection)

Section	Document	Reference Section Title
3.5.2.2.1	OPG.02	Network (*Collecting radio network information, e.g. cell change notification, measurement reports etc. for mobility decisions)

2.1.3 Procedures

The following procedures defined in 3GPP shall apply to the OP Network Events capabilities:

NF	Section	TS	Procedure Name
NEF/SCEF	4.11.6.3	3GPP 23.502	Nnef_EventExposure_Subscribe (*Configuration of monitoring events for common network exposure)

2.1.4 API

The following API as defined in 3GPP shall be endorsed for Operator Platform

NF	Section	TS	API name
SCEF	4.4.2	29.122	Procedures over T8 reference point / Monitoring Procedures) (*Loss of connectivity; UE reachability; Location Reporting; Change of IMSI-IMEI(SV) Association; Roaming Status; Communication Failure; PDN connectivity status; Availability after DDN Failure; API support capability.)
SCEF	Annex A (pp. 323)	29.122	Monitoring Event API - YAML

The following API parameters have been identified as missing in 3GPP

NF	Section	TS	API name
NEF	4.4.2	29.522	NEF Northbound Interface / Procedures for Monitoring (*29.122 emphasizes SCEF-T8-Procedures can be applied in the NEF as well)

2.2 QoS Management API

2.2.1 Description

The purpose of the Quality of Service (QoS) Management API is to enable an OP the ability to expose the network driven capabilities related to 5G QoS Identifiers (5QI). With this API, the OP provides to 3rd party consumer the capability to adapt the quality and capacity of the network based on its application requirements. This kind of customization can be applied to many Business-to-Business (V2X drone control, etc.) and Business-to-Consumer (immersive communications, gaming, etc.) use cases where the quality can be adapted and configured in the network to guarantee the best user experience.

For this API definition it makes sense to reuse existing 3GPP NEF/SCEF API for QoS management. The corresponding API in the NBI interface will be defined by the Linux

Foundation's CAMARA project considering mapping to the references included in the following sections.

2.2.2 Requirement and Service Aspects

The requirements for the API are specified in the GSMA PRD OPG.02 [1]. in the following section(s).

Section	Document	Reference Section Title
5.1.4	OPG.02	Southbound Interface to Network Resources

2.2.3 Procedures

The following procedures defined in 3GPP shall apply to the OP's QoS Management capabilities:

NF	Section	TS	Procedure Name
NEF	4.15.6.6/6a	23.502	Setting up an Application Function (AF) session with required QoS procedure
SCEF	4.4.13	29.122	Procedures for setting up an Application Server (AS) session with required QoS
NEF	4.4.9	29.522	Procedures for setting up an AF session with required QoS

NOTE: The AsSessionWithQoS API is define in the NEF 3GPP spec as a reused API (section 5.3 TS 29.522). It adapts the development and definition from the SCEF API (4G architecture) to the NEF (5G Architecture).

2.2.4 API

The following API as defined in 3GPP shall be endorsed for Operator Platform.

NF	Section	TS	API name
NEF	5.3	29.522	AsSessionWithQoS (Reused APIs Table)
SCEF	5.14	29.122	AsSessionWithQoS

2.3 Traffic Influence API

2.3.1 Description

The purpose of the Traffic Influence API is to enable OP the ability to expose the network driven capabilities to influence on the traffic transport paths between the UE and the Application Server hosted at the Cloud Resources. The selected API is based on 3GPP definitions. Some modifications the API would be required in the East-West Bound Interface (EWBI) interface to be usable in the federated environment. The corresponding API in the NBI interface will be defined by the Linux Foundation's CAMARA project's specifications for the Service APIs in the NBI interface.

2.3.2 Requirement and Service Aspects

The requirements for the API are specified in the GSMA PRD OPG.02 [1] in the following section(s).

Section	Document	Reference Section Title
5.1.4	OPG.02	Southbound Interface to Network Resources

2.3.3 Procedures

The following procedures defined in 3GPP shall apply to the OP Traffic Influence capabilities.

NF	Section	TS	Procedure Name
NEF	4.3.6	23.502 R 17	Application Function influence on traffic routing
NEF	5.2.6.7	23.502 R 17	Nnef_TrafficInfluence Service
NEF	5.4	23.502 R 17	TrafficInfluence
NEF	4.4.7	29.522 R 17	Procedures for Traffic Influence

2.3.4 API

The following API as defined in 3GPP shall be endorsed for Operator Platform.

NF	Section	TS	API name
NEF	5.4	29.522 R 17	TrafficInfluence

2.4 Application Relocation API

2.4.1 Description

The purpose of the Application Relocation API is to enable the OP to manage and maintain inter- and intra-connectivity between applications as they move between different cloud and network resources. This concept is aligned with the architecture and definitions descrired in 3GPP TS 23.558 [4]. .

2.4.2 Requirement and Service Aspects

The requirements for the API are specified in the GSMA PRD OPG.02 [1]. in the following section(s).

Section	Document	Reference Section Title
4.7	OPG.02	Applications deployment in the Federated Operator Domain (Figure 15 and text)
5.1.6.2.6	OPG.02	Service Provisioning
5.2.2.3.6	OPG.02	Mobility Enforcements
5.2.2.6.3	OPG.02	Network and OP responsibilities for application session continuity
5.2.2.6.5	OPG.02	Edge Applications Role in Session Continuity Process
Annex F	OPG.02	5G Core Network Application Session Continuity Enabler Services

2.4.3 Procedures

The following procedures defined in 3GPP shall apply to the Application Relocation capabilities:

NF	Section	TS	Procedure Name
	8.8	23.558	Service Continuity
EES	5.8	29.558	Eees_ACRManagementEvent Service
EES	5.9	29.558	Eees_AppContextRelocation Service
EES	5.11	29.558	Eees_EELManagedACR Service
EES	5.12	29.558	Eees_ACRStatusUpdate Service
EES	5.8.2.3	29.558	Eees_ACRManagementEvent_UpdateSubscription
EES	5.8.2.4	29.558	Eees_ACRManagementEvent_Unsubscribe
EES	5.8.2.5	29.558	Eees_ACRManagementEvent_Notify
EES	5.9.2.3	29.558	Eees_AppContextRelocation_ACRDetermination_Request
EES	5.11.2.2	29.558	Eees_EELManagedACR_Request
EES	5.11.2.3	29.558	Eees_EELManagedACR_Subscribe
EES	5.11.2.4	29.558	Eees_EELManagedACR_Notify
EES	5.12.2.2	29.558	Eees_ACRStatusUpdate_Request

2.4.4 API

The following API as defined in 3GPP shall be endorsed for Operator Platform.

NF	Section	TS	API name
EES	8.7	29.558	Eees_ACRManagementEvent API
EES	8.9	29.558	Eees_EECContextRelocation API
EES	8.10	29.558	Eees_EELManagedACR API
EES	8.11	29.558	Eees_ACRStatusUpdate API

2.5 Confirm User Location API

2.5.1 Description

The purpose of the Confirm User Location API is to enable an OP with the ability to provide user location information to consumers. This can be used in combination with other network driven capabilities to trigger configurations within the network to provide services based on the location of the final users (e.g., to deploy resources and applications closer to the user so that latency can be reduced to the minimum and improve the user experience).

2.5.2 Requirement and Service Aspect

The requirements for the API are specified in the GSMA PRD OPG.02 [1]. in the following section(s).

Section	Document	Reference Section Title
3.5.2.2.1	OPG.02	UC Location Retrieval

5.1.6.2.4	OPG.02	Cloudlet selection

2.5.3 Procedures

The following procedures defined in 3GPP shall apply to the OP's user location information capabilities.

NF	Section	TS	Procedure Name
NEF	4.4.2	29.522	Procedures for Monitoring
SCEF	4.4.2	29.122	Monitoring Procedures
EES	8.6.2	23.558	UE Location

2.5.4 API

The following API as defined in 3GPP shall be endorsed for Operator Platform

NF	Section	TS	API name
SCEF	5.3	29.122	MonitoringEvent
NEF	5.3	29.522	MonitoringEvent (Reused APIs Table)
EES	8.2	29.558	Eees_UELocation

2.6 Mobility Triggers API

2.6.1 Description

This API provides the required support from the network so that the OP can be informed about the need to move an application session to a different anchor point or of the actual move. A possible use case is when the user moves to a new location. By means of this API, the network alerts the OP about the change, so that the OP can perform the required new configurations for the user to access the applications in the new location.

2.6.2 Requirement and Service Aspects

The requirements for the API are specified in the GSMA PRD OPG.02 [1]. in the following section(s).

Section	Document	Reference Section Title
2.2.7	OPG.02	Mobility Requirements (*5G Connectivity Models for Edge; *Roaming Requirements; *Geographic Conditions on Mobility)
5.2.2.3.2	OPG.02	Mobility Management

2.6.3 Procedures

The following procedures defined in 3GPP shall apply to the Mobility Management capabilities.

NF	Section	TS	Procedure Name
NEF	4.4.2	29.522	Procedures for Monitoring
SCEF	4.4.2	29.122	Monitoring Procedures

2.6.4 API

The following API as defined in 3GPP shall be endorsed for Operator platform.

NF	Section	TS	API name
NEF	5.3	29.522	MonitoringEvent API (Reused API Table)
SCEF	5.3	29.122	MonitoringEvent API

These APIs, defined in 3GPP, provides support to retrieve information related to different events related to the OP's Mobility Triggers concept. For instance:

- LOSS_OF_CONNECTIVITY
- UE_REACHABILITY
- LOCATION_REPORTING
- AREA_OF_INTEREST

2.7 Mobility Control API

2.7.1 Description

The Mobility Control API is responsible for controlling when an application session is to be moved to a different anchor point. For instance, due to a failure an application relocation occurs. After relocating an application, a change of anchor point may be required, if this is the case, the OP can use this API to adjust in the network the most appropriate configuration to access the application new endpoint.

2.7.2 Requirements and Service Aspects

The requirements for the API is specified in the GSMA PRD OPG.02 [1]. in the following sections:

Section	Document	Reference Section Title
2.2.7.3	OPG.02	Requirements for Application Session Continuity

2.7.3 Procedures

The following procedures defined in 3GPP shall apply to the OP's traffic influence capabilities:

NF	Section	TS	Procedure Name
NEF	4.3.6	23.502 R 17	Application Function influence on traffic routing
NEF	5.2.6.7	23.502 R 17	Nnef_TrafficInfluence Service
NEF	5.4	23.502 R 17	TrafficInfluence
NEF	4.4.7	29.522 R 17	Procedures for Traffic Influence

2.7.4 API

The following API as defined in 3GPP shall be endorsed for Operator Platform

NF	Section	TS	API name
NEF	5.4	29.522 R 17	TrafficInfluence

3 Network key enabling capabilities

3.1 End user Public IP address to MSISDN mapping.

3.1.1 Introduction

Whilst the Mobile Subscriber Integrated Services Digital Network Number (MSISDN) is a valid personal identifier, the end user will unlikely consent this information to be shared with every Application Provider for privacy reasons. Therefore, other identifiers are required to refer to a subscriber in the Northbound API requests. The subscriber's Public IP address and port as used to communicate with the Application Provider's Application Backend could be one of those identifiers. If used, the OP should be able to map this Public IP address and port to an identifier that it can use to identify the subscriber on its Southbound Interfaces, e.g. the MSISDN. Because no standardised solutions exist to do this mapping, this section defines a new API that networks can provide for that purpose and suggest ways to realise the functionality exposed by that API based on enablers that are commonly used in networks.

As an illustration, the Service API for obtaining an Anonymised Subscriber Identifier as proposed in Linux Foundation's CAMARA project enables an application to request an end user identity to the network by providing the end device's public IP address. To realise that API with the Operator Platform, the OP should be able to interact with the network to map that end device's public IP address into its MSISDN for the Operator Platform to return an anonymised user identity that could be used by the application.

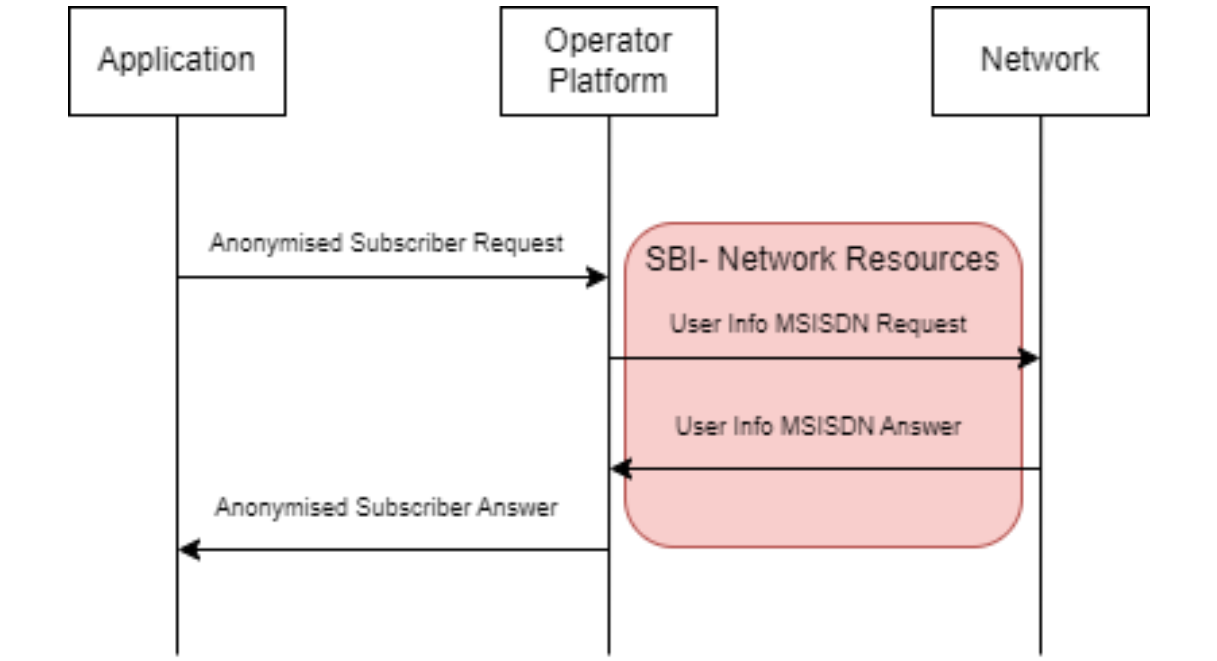

Figure 1: Service flow for an API relying on IP address mapping (e.g. to obtain an anonymised subscriber identity)

3.1.2 Network setup

This section is meant to introduce mechanisms intended to support operators with implementations at the network level that will facilitate to obtain information related to users.

3.1.2.1 Deterministic NAT

Network Address Translation (NAT) is used in order to map the a UE's Private IP address to a Public IP address for communication to services on the public internet. If deterministic NAT is used, each private UE IP is mapped to a specific port range of a public IP address of the Carrier Grade NAT (CGNAT). This allows to use this mapping also in the reverse direction and map a public IP address and port combination back to the Private IP address and thus to the UE for which it would then be possible to determine the MSISDN.

Deterministic NAT requires that the ratio between the range of private IP addresses and public IP addresses available is sufficient to ensure that the port range allocated for each UE Private IP address can satisfy the user's needs in terms of concurrent connections.

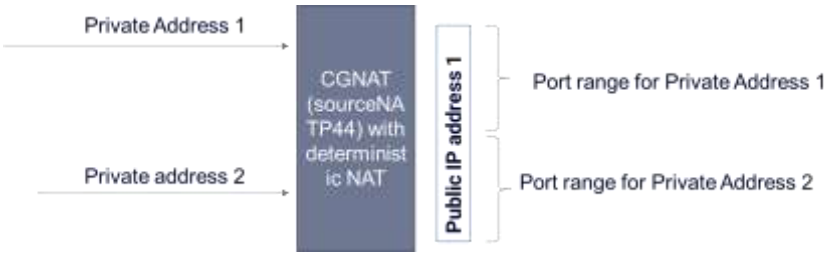

Figure 2: Deterministic NAT Solution

NOTE: sourceNATP44 is described in RFC 7857 "Updates to Network Address Translation (NAT) Behavioral Requirements".

3.1.2.2 End user directory

The network can provide a user directory where IP address (and port) mappings will be stored along with the corresponding user MSISDN. This directory could then be populated during the User Equipment's connectivity to the network and consulted when a public IP address needs to be mapped to an MSISDN. Such directory is not available today within a standard 3GPP core network architecture but could be considered and supported in functions such as the Unified Data Management (UDM).

3.2 User Info API

This API allows to request the operator platform for an identifier for a subscriber based on their public IP address and port.

Operation	HTTP Methods	Resource URI	Qualifier
IdentifyUser	GET	/naas/networkresources/v1	M

Table 2: User Info API: Operations

3.2.1 GET Method: IdentifyUser

The GET method enables to retrieve user identifier information from the network, such as the MSISDN by providing a public IP and port.

Parameter Name	P	Cardinality	Description
publicIPAddress	M	1	Origin public IP address from which the UE is connecting to an application backend.
port	M	1	Port from which the UE is connecting to an application backend.
protocol	O	1	Transport protocol (e.g UDP, TCP, SCTP)
identityType	O	1	Type of identifiers that need to be retrieve e.g MSISDN, Private IP

Table 3: Request Parameters

Parameter Name	P	Response Codes	Description
identifier	M	200	User MSISDN, Private IP, other identifiers
errorResponse	C	400	Bad Request.
errorResponse	C	401	Unauthorized
errorResponse	C	403	Permissions
errorResponse	C	404	Not Found
errorResponse	C	405	Method not allowed
errorResponse	C	406	Not Acceptable
errorResponse	C	429	Too many requests
errorResponse	C	500	Internal Server Error

Parameter Name	P	Response Codes	Description
errorResponse	C	502	Bad Gateway
errorResponse	C	503	Service Unavailable.
errorResponse	C	504	Request time exceeded

Table 4: Response Parameters

Note: Responses are for immediate use only and their validity cannot be guaranteed over a longer time.

3.2.2 Data Model

3.2.2.1 Simple data types and enumerations

This subclause defines simple data types and enumerations that can be referenced from data structures defined in the previous subclauses.

Attribute Name	Data Type	Description
publicIPAddress	String	Public IPaddress (IPv4 or IPv6).
port	Integer	A port number assigned to uniquely identify a connection endpoint and to direct data to a specific service.
protocol	String	Transport protocol (e.g UDP, TCP, SCTP)
identityType	String	Type of identifiers that need to be retrieve e.g MSISDN, Private IP
identifier	Object	User identity information that contains: type: (e.g MSISDN, Private IP) id: Identifier value. In case of 'MSISDN' type, 'E164 with +' format must be used.

Table 5: User Info API data types

Annex A Open API definitions

A.1 User Info API

```
openapi: 3.0.3
info:
  version: '1.0.0'
  title: 'User Info'
  description: |
    # Introduction
    ---
    RESTful API that allows an OP to map the UE origin Public IP address and port to an identifier that it can use to identify the subscriber on its Southbound Interfaces (e.g. the MSISDN, Private IP)

    ---
    # API Scope

    ---
```

```
    APIs defined in this version of the specification can be categorized into the following areas:

    * __UserInfo__ - To retrieve user identifier associated to some origin connection information such as public IP
address and port.

    ---
    # Definitions
    ---
    * __publicIPAddress__ - Origin public IP address from which the UE is connecting an application backend.
    * __port__ - Port from which the UE is connecting to an application backend.
    * __protocol__ - Transport protocol (e.g. U DP, TCP, SCTP).
    * __identityType__ - Type of identifiers that need to be retrieve e.g MSISDN, Private IP.
    * __identifier__ - User MSISDN, Private IP or other possible identifiers.

    ---
    # API Operations
    ---

    __UserInfo__
    * __IdentifyUser__ - Retrieve a user identifier associated to some origin connection information.
    © 2023 GSM Association.
    All rights reserved.

externalDocs:
  description: GSMA, SBI-Network Resources APIs
  url: 'https://www.gsma.com/futurenetworks/5g-operator-platform/'
servers:
  - url: '{apiRoot}/naas/networkresources/v1'
    variables:
      apiRoot:
        default: https://operatorplatform.com

security:
  - oAuth2ClientCredentials:
    - net-resources

components:
  securitySchemes:
    oAuth2ClientCredentials:
      type: oauth2
      description: This API uses OAuth 2 with the client credentials grant flow.
      flows:
        clientCredentials:
          tokenUrl: '/oauth2/token'
          scopes:
            net-resources: Access to the Network Resources APIs

  schemas:
    IdentifierMSISDN:
      type: object
      required:
        - type
        - id
      properties:
        type:
          type: string
          description: Type of the user identifier retrieved (MSISDN).
          example: "msisdn"
```

```
        id:
          type: string
          description: MSISDN value. 'E164 with +' format must be used [+] [country code] [subscriber number including area code] and can have a maximum of fifteen digits..
          example: '+346667778889'
          format: ^\+[1-9]\d{1,14}$

    IdentifierPrivateIP:
      type: object
      required:
        - type
        - id
      properties:
        type:
          type: string
          description: Type of the user identifier retrieved (Private IP address).
          example: "private IP address"
        id:
          type: string
          description: Private IP address of the UE.
          example: '192.168.0.20'
          format: ^(127(?:\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)){3}$)|(10(?:\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)){3}$)|(192\.168(?:\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)){2}$)|(172\.(?:1[6-9]|2\d|3[0-1])(?:\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)){2}$)

    errorResponse:
      type: object
      properties:
        code:
          type: string
          description: A short, human-readable summary of the problem type
        status:
          type: integer
          description: The HTTP status code
        message:
          type: string
          description: This parameter appears when there was an error. Human readable explanation specific to this occurrence of the problem

    UserInfoMSISDNResponse:
      type: object
      required:
        - identifier
      properties:
        identifier:
          $ref: '#/components/schemas/IdentifierMSISDN'

    UserInfoPrivateIPResponse:
      type: object
      required:
        - identifier
      properties:
        identifier:
          $ref: '#/components/schemas/IdentifierPrivateIP'

  responses:
    400BadRequest:
      description: Bad Request
```

```
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/errorResponse'
        examples:
          InvalidIP:
            value :
              {
                "code": "INVALID_ARGUMENT",
                "status": 400,
                "message": "Invalid or missing IP header"
              }
          InvalidPort:
            value :
              {
                "code": "INVALID_ARGUMENT",
                "status": 400,
                "message": "Invalid or missing Port header"
              }
    401Unauthorized:
      description: Unauthorized
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          examples:
            InvalidCredentials:
              value:
                {
                  "code": "UNAUTHENTICATED",
                  "status": 401,
                  "message": "Request not authenticated due to missing, invalid, or expired credentials"
                }

    403Forbidden:
      description: Forbidden
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          examples:
            InsufficientPermissions:
              value:
                {
                  "code": "PERMISSION_DENIED",
                  "status": 403,
                  "message": "Client does not have sufficient permissions to perform this action"
                }

    404NotFound:
      description: Subscriber Not Found
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          examples:
            SubscriberNotFound:
```

```
            description: The specified IP address and port are not currently associated with any customers of this
service
            value:
              {
                "code": "NOT_FOUND",
                "status": 404,
                "message": "No subscriber found with the specified IP address and port"
              }

    405MethodNotAllowed:
      description: Method Not Allowed
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          examples:
            MethodNotAllowed:
              description: An HTTP verb other than GET has been used to try and access the resource
              value:
                {
                  "code": "METHOD_NOT_ALLOWED",
                  "status": 405,
                  "message": "The request method is not supported by this resource"
                }

    406Unacceptable:
      description: Not Acceptable
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          examples:
            NotAcceptable:
              description: A response format other than JSON has been requested
              value:
                {
                  "code": "NOT_ACCEPTABLE",
                  "status": 406,
                  "message": "The server cannot produce a response matching the content requested by the client
through Accept-* headers"
                }

    429TooManyRequests:
      description: Too Many Requests
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          examples:
            TooManyRequests:
              description: Access to the API has been temporarily blocked due to quota or spike arrest limits being
reached
              value:
                {
                  "code": "TOO_MANY_REQUESTS",
                  "status": 429,
                  "message": "Either out of resource quota or reaching rate limiting"
                }
```

```
    500InternalServerError:
      description: Internal Server Error
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          example:
            {
              "code": "INTERNAL",
              "status": 500,
              "message": "The service is currently not available"
            }

    502BadGateway:
      description: Bad Gateway
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          example:
            {
              "code": "BAD_GATEWAY",
              "status": 502,
              "message": "The service is currently not available"
            }

    503ServiceUnavailable:
      description: Service Unavailable
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          example:
            {
              "code": "UNAVAILABLE",
              "status": 503,
              "message": "The service is currently not available"
            }

    504GatewayTimeout:
      description: Gateway Time-Out
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/errorResponse'
          example:
            {
              "code": "TIMEOUT",
              "status": 504,
              "message": "The service is currently not available"
            }


paths:
  /identifyUser:
    get:
      summary: Retrieve a user identifier given a public IP and Port
```

```
    tags:
      - UserInfo
    parameters:
      - in: header
        name: publicIPAddress
        description: Origin public IP address from which the UE is connecting to an application backend.
        required: true
        examples:
          ipv4:
            value: "84.125.93.10"
          ipv6:
            value: "2001:db8:85a3:8d3:1319:8a2e:370:7344"
        schema:
          type: string
          format: ipv4/ipv6
          oneOf:
            - pattern: '^(([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-5])\.){3}([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-
5])$'
            - pattern: '^((:|(0?|([1-9a-f][0-9a-f]{0,3}))):)((0?|([1-9a-f][0-9a-f]{0,3})):){0,6}(:|(0?|([1-9a-f][0-9a-f]{0,3})))$'

      - in: header
        name: port
        description: Port from which the UE is connecting to an application backend.
        required: true
        example: 20000
        schema:
          type: integer
          minimum: 1024
          maximum: 65535

      - in: header
        name: protocol
        description: The transport protocol in use.
        required: false
        examples:
          TCP:
            value: "tcp"
          UDP:
            value: "udp"
          SCTP:
            value: "sctp"
        schema:
          type: string
          default: tcp

      - in: header
        name: identityType
        description: Type of identifiers that need to be retrieve.
        required: false
        examples:
          MSISDN:
            value: "msisdn"
          Private IP address:
            value: "private ip"
        schema:
          type: string
          default: msisdn
```

```
    responses:
      "200":
        description: User identified successfully
        content:
          application/json:
            schema:
              oneOf:
                - $ref: '#/components/schemas/UserInfoMSISDNResponse'
                - $ref: '#/components/schemas/UserInfoPrivateIPResponse'


      "400":
        $ref: '#/components/responses/400BadRequest'
      "401":
        $ref: '#/components/responses/401Unauthorized'
      "403":
        $ref: '#/components/responses/403Forbidden'
      "404":
        $ref: '#/components/responses/404NotFound'
      "405":
        $ref: '#/components/responses/405MethodNotAllowed'
      "406":
        $ref: '#/components/responses/406Unacceptable'
      "429":
        $ref: '#/components/responses/429TooManyRequests'
      "500":
        $ref: '#/components/responses/500InternalServerError'
      "502":
        $ref: '#/components/responses/502BadGateway'
      "503":
        $ref: '#/components/responses/503ServiceUnavailable'
      "504":
        $ref: '#/components/responses/504GatewayTimeout'
```

Annex B Document Management

B.1 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	03 Oct 2022	New PRD defining the Soutbound Interface of the Operator Platform to the network resources	ISAG	Milan Lalovic / British Telecom
2.0	29 Mar 2023	Update implementing OPG.03 CR1002 introducing Public IP address mapping	ISAG	Tom Van Pelt / GSMA

B.2 Other Information

Type	Description
Document Owner	Operator Platform Group
Editor / Company	Milan Lalovic / British Telecom

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at prd@gsma.com

Your comments or suggestions & questions are always welcome.