

Microsoft MBIM Extensions for 5G

Interface specification

This document specifies extensions to Mobile Broadband Interface Model (MBIM) for 5th generation mobile network

Revision 1.17

DOCUMENT UPDATE HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Revision | Author | Changes |
| 03/26/2019 | 1.00 | Microsoft | 5G NSA (Option 3) extensions in Section 2. |
| 12/12/2019 | 1.06 | Microsoft | Draft of MBIM extensions for 5G NGC features for 21H1 (Windows Fe) commercialization in Sections 3 and Appendix. |
| 01/20/2020 | 1.07 | Microsoft | Updates based on review comments on Rev 1.06 |
| 01/29/2020 | 1.08 | Microsoft | Updated based on partner comments on Rev 1.07 |
| 01/31/2020 | 1.09 | Microsoft | Updated the informative sequence diagrams in 4.2 |
| 03/12/2020 | 1.10 | Microsoft | Editorial and clarifying changes. |
| 03/17/2020 | 1.11 | Microsoft | Updated WAKE\_REASON response and its TLV data structure; Moved the definition of MBIM\_CID\_MS\_NETWORK\_PARAMS out of Section 3. |
| 03/24/2020 | 1.12 | Microsoft | Replace NTimingAdvanceOffset with TimingAdvance in table 3.12-3 MBIM\_NR\_SERVING\_CELLS of MBIM\_CID\_BASE\_STATIONS\_INFO section. |
| 03/25/2020 | 1.13 | Microsoft | Added Section 3.13 for Modified MBIM\_CID\_MS\_LTE\_ATTACH\_STATUS |
| 04/30/2020 | 1.14 | Microsoft | Clarification on parameter ContextType in MBIM\_CID\_CONNECT |
| 7/24/2020 | 1.15 | Microsoft | Typo fixings in data structures MBIM\_DEVICE\_CAPS\_INFO\_V3 and MBIM\_SUBSCRIBER\_READY\_INFO\_EX3 |
| 10/12/2020 | 1.16 | Microsoft | Minor clarifications and typo fixing in Section 3.9.6 for data structure MBIM\_CONNECT\_INFO\_EX3 |
| 10/22/2020 | 1.17 | Microsoft | Clarification on encoding for the structure MBIM\_MS\_PLMN in Table 4.1‑2; Clarification for the field DataSubClass in the structure MBIM\_DEVICE\_CAPS\_INFO\_V3 in 3.3.4 |

# Description

To fulfill the operator’s requirements for 5G NR networks and enhanced 5G experiences, the Mobile Broadband Interface Model (MBIM) 1.0 Errata-1 Interface shall be extended. This specification defines the extensions.

The Microsoft MBIM extensions are in the form of new MBIM Extensions Releases, modified existing CIDs and data structures, as well as new CIDs and new data structures.

Section 2 of this document, *MBIM Interface Extensions for 5G NSA (Option 3)*, defines the MBIM extensions for 5G Option 3, commonly referred to as 5G NSA. In this option, the core network is the EPC.

Section 3 of this document, MBIM Interface Extensions for 5G NGC – Phase 1*,* defines the MBIM extensions for first part of features for 5G Options 2, 4, 7 and 5. In these 5G options, the core network is the 5G Next Generation Core (NGC, aka 5GC). This part of features for 5G NGC are those intended for commercialization in 2021H1 in Windows Fe release. Refer to this section for details on the contents for those features.

Section 5 of this document, Appendix, contains definitions of common data structures, informative sequence diagrams of usage of 5G NGC extensions, clarification on status code MBIM\_STATUS\_BUSY, and a list of general non-5G-specific improvements that are made throughout this document.

Figure 1 summarizes the characteristics of the seven 5G-LTE deployment options.

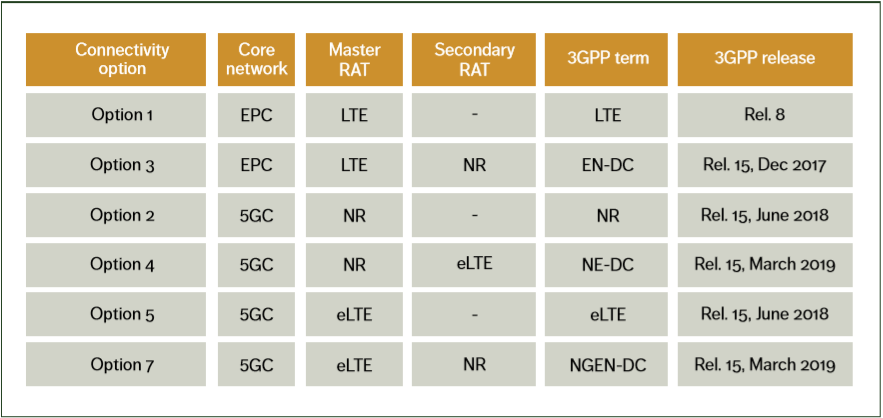


Figure 1: 5G Deployment Options

5G NSA in general includes Options 3, 4 and 7[13]. But in this document, 5G NSA refers only to Option 3, the quickest 5G NSA deployment option that leverage LTE for 5G eMBB use case. On the other hand, although Options 4 and 7 are NSA cases, they are supported together with Option 2 (the most common 5G SA case) and Option 5 (potentially less common 5G SA case) with the extensions in Section 3. They have a common characteristic of using 5GC core network.

# MBIM Interface Extensions for 5G NSA (Option 3)

The MBIM 1.0 Errata Interface Specification[1] has a mechanism to add and advertise optional CIDs, but it lacks a mechanism to change the existing CIDs (new payloads or modified payload) or to introduce changes in any aspect that cannot be accommodated by optional CIDs. Each payload in [1] may consist of fixed sized members or dynamic sized (offset/size pairs) members. If a dynamic sized member exists, then the last member has variable size buffer. [1] defines a fixed location for this buffer and therefore new members cannot be added before it.

To introduce new member(s) for the existing CID payload, it means a breaking change. The breaking change means that the existing payload is not anymore backward compatible with the new payload.

[1] defines MBIM Release number 1.0 and MBIM Extensions Release number 1.0, in Section 6.4 “MBIM EXTENDED FUNCTIONAL DESCRIPTOR” and Section 6.5 “MBIM EXTENDED FUNCTIONAL DESCRIPTOR”. For USB-based MBIM devices, the devices use the USB descriptors to advertise its MBIM Release number and MBIM Extensions Release number it conforms to. For MBIM devices based on other bus types that use MbbCx APIs, an API is provided for the device to advertise its MBIM Release number and MBIM Extensions Release number.

Note that [1] does not define any way for host to advertise its MBIM Release number and MBIM Extensions Release number.

This spec introduces a new MBIM Extensions Release number 2.0 for a set of breaking changes under existing MBIM Release number 1.0. This spec also adds an optional CID for host to advertise its MBIM Release number and Extensions Release number to MBIM devices. Like MBIM Extensions Release number 1.0, MBIM Extensions Release number 2.0 is under MBIM Release number 1.0. Unless explicitly mentioned and modified, all unmentioned payloads, CIDs and procedures in MBIM Extensions Release number 1.0 carry over and stay unchanged in MBIM Extensions Release number 2.0.

Section 1 discusses how host and device exchange version information. It also discusses how and what backward compatibility is supportable between hosts and devices with various versions. Although only MBIM Extensions Release number 2.0 is introduced in this spec, the versioning scheme defined in this spec is designed to accommodate additional versions in the future. Section 2.2 defines the optional MBIM\_CID\_VERSION. Sections 2.3 through 2.6 specifies the updated mandatory CIDs from [1].

## Versioning Scheme

In this section, the term “MBIMEx version” refers to the MBIM Extensions Release number.

The host gets to know a device’s MBIMEx version via two ways: 1) MBIM EXTENDED FUNCTIONAL DESCRIPTOR, and 2) the optional MBIM\_CID\_VERSION if device supports it and declares the support for it. If these two are different, the higher one governs as the device’s MBIMEx version for the duration the device stays enumerated to host. The higher MBIMEx version will be referred to as the device’s announced MBIMEx version. A device’s announced MBIMEx version can be lower than its native MBIMEx version (the highest MBIMEx version the device can support). The necessity of that will become apparent later when discussing the compatibility matrix.

The device gets to know a host’s MBIMEx version explicitly only via MBIM\_CID\_VERSION defined in 2.2.

Host in any release always queries device for supported services and supported CIDs using MBIM\_CID\_DEVICE\_SERVICES at the beginning of device initialization sequence. If a device supports MBIM\_CID\_VERSION and advertises its support in MBIM\_CID\_DEVICE\_SERVICES query response, then

* a host that does not understand MBIM\_CID\_VERSION or has an MBIMEx version lower than 2.0 shall ignore it.
* a host that understands MBIM\_CID\_VERSION and has native MBIMEx version 2.0 or higher shall send MBIM\_CID\_VERSION to device with host’s native MBIMEx version, and the CID shall be the first CID to device after receiving MBIM\_CID\_DEVICE\_SERVICES response.

MBIM\_OPEN\_MSG, MBIM\_CID\_DEVICE\_CAPS and MBIM\_CID\_DEVICE\_SERVICES will be sent before MBIM\_CID\_VERSION, so they and their response cannot change in different version. If device need handle other commands or responses that have different version, the device need make decision based on if it receives MBIM\_CID\_VERSION from host. If device received MBIM\_CID\_VERSION from host, device knows the host’s native MBIMEx version. If device didn’t receive MBIM\_CID\_VERSION from host, device shall assume that host’s native MBIMEx version is 1.0.

Feature-wise, a higher MBIMEx version must be a superset of all lower MBIMEx versions. A host shall support all devices with an announced MBIMEx version at or below the host’s native MBIMEx version. If a device’s announced MBIMEx version is higher than a host’s native MBIMEx version, the host is not expected to support the device and the exact behavior of the host in this situation is undefined.

A device with intention to work with older hosts should initially advertise MBIMEx version 1.0 (or the lowest host MBIMEx version that the device is intended to work with) in MBIM extended functional descriptor. If the host sends MBIM\_CID\_VERSION and the host has a higher MBIMEx version than device initially advertises, the device should, in the MBIM\_CID\_VERSION response, indicates a higher MBIMEx version, up to the smaller between the host’s native MBIMEx version and the device’s native MBIMEx version.

Example 1: A device can support MBIMEx version 2.0, but it is intended to work with older OS’s that do not support MBIMEx 2.0. The device initially advertises MBIMEx version 1.0 in USB descriptors and advertises the support for optional MBIM\_CID\_VERSION. When inserted into a host running RS4 OS, the host does not understand MBIM\_CID\_VERSION and will not send MBIM\_CID\_VERSION to device. To host, the device’s MBIMEx version is 1.0. The host will continue to send other CIDs in the initialization sequence. Upon receiving other CIDs than MBIM\_CID\_VERSION, the device knows the host is with MBIMEx 1.0. Both sides proceed to conform to MBIMEx 1.0. On the other hand, when the same device is inserted in a host running 19H1 OS with native MBIMEx version 2.0, the host sends MBIM\_CID\_VERSION to device informing device of host’s MBIMEx version 2.0. The device sends back MBIM\_CID\_VERSION response with the device’s announced MBIMEx version 2.0. From there, both sides proceed to conform to MBIMEx 2.0.

As with MBIMEx version 1.0, optional CIDs may be introduced in MBIMEx version 2.0. New optional CIDs will not change an MBIMEx version. A device is allowed to send notification of any optional CID. If a host does not understand a CID, the host shall ignore it. The host knows if a particular optional CID is supported by device via MBIM\_CID\_DEVICE\_SERVICES query.

Example 2: A device supports MBIMEx version 3.0. It requires users to upgrade OS to 19H2 with MBIMEx version 3.0 support and it has clearly declared so. The device is inserted to a host running RS5 that supports only MBIMEx version 1.0. The device advertises the MBIMEx version 3.0 in its USB Extended Descriptor. The host doesn’t recognize the version and does not understand MBIM\_CID\_VERSION. The host does not support such a device and the device functionality does not work.

Example 3: A device supports MBIMEx version 3.0 and is inserted to a host running OS 19H1 supporting MBIMEx version 2.0. The device advertises the MBIMEx version 1.0 in its USB Extended Descriptor. The host checks if the device supports the optional CID\_MBIM\_VERSION. The device supports it and the host sends CID\_MBIM\_VERSION QUERY with payload MBIMEx version 2.0. Seeing the host supports MBIMEx 2.0, the device responds with CID\_MBIM\_VERSION QUERY response announcing MBIMEx version 2.0. Both sides proceed with MBIMEx version 2.0

Example 4: A device supports MBIMEx version 2.0 and is inserted to a host supporting MBIMEx version 3.0. The device advertises the MBIMEx version 1.0 in its USB Extended Descriptor. The host checks if the device supports the optional CID\_MBIM\_VERSION. The device supports it and the host sends CID\_MBIM\_VERSION QUERY with payload = 3.0. The device sends back CID\_MBIM\_VERSION QUERY response with payload = 2.0. Both sides proceed with MBIMEx version 2.0.

The table below shows a compatibility matrix with three hypothetical hosts and three hypothetical device s, with their native MBIMEx version stated. The devices advertise MBIMEx version 1.0 initially in USB descriptor. The matrix shows how each of the devices may behave with each of the hosts.

|  |  |  |  |
| --- | --- | --- | --- |
| OS  Modem | RS5 or earlier  (native MBIMEx version 1.0) | 19H1  (native MBIMEx version 2.0) | 21H1  (native MBIMEx version 3.0) |
| 4G device  Native MBIMEx version 1.0 | Device initially advertises 1.0. No MBIM\_CID\_VERSION exchange.  Compatible device & host – works by default with MBIMEx version 1.0 | Device initially advertises 1.0. No MBIM\_CID\_VERSION exchange. The host works with the device with MBIMEx version 1.0 spec. | Device initially advertises 1.0. No MBIM\_CID\_VERSION exchange. The host works with the device with MBIMEx version 1.0 spec. |
| 5G-NSA device  Native MBIMEx version 2.0 | Device initially advertises 1.0. No MBIM\_CID\_VERSION exchange. Device knows host having 1.0 and proceeds with  MBIMEx version 1.0 | Device initially advertises 1.0. Host sends MBIM\_CID\_VERSION to tell host supports 2.0. Device responds with 2.0. Both sides proceed with 2.0. | Device initially advertises 1.0. Host sends MBIM\_CID\_VERSION to tell host supports 3.0. Device responds with 2.0. Both sides proceed with 2.0. |
| 5G-SA device  Native MBIMEx version 3.0 | Device initially advertises 1.0. No MBIM\_CID\_VERSION exchange. Device knows host having 1.0 and proceeds with  MBIMEx version 1.0 | Device initially advertises 1.0. Host sends MBIM\_CID\_VERSION to tell host supports 2.0. Device responds with 2.0. Both sides proceed with 2.0. | Device initially advertises 1.0. Host sends MBIM\_CID\_VERSION to tell host supports 3.0. Device responds with 3.0. Both sides proceed with 3.0. |

The table below lists all existing CIDs that are modified in MBIMEx version 2.0 and their modified payloads. All unmentioned payloads in these CIDs and all other CIDs not mentioned in the table below carry over from MBIMEx version 1.0 and remain unchanged.

Table 2.1‑1: Modified CIDs and payloads in MBIMEx version 2.0

|  |  |
| --- | --- |
| CID | Payload |
| MBIM\_CID\_MS\_WAKE\_REASON  MBIM\_CID\_REGISTER\_STATE | MBIM\_REGISTRATION\_STATE\_INFO\_V2 |
| MBIM\_CID\_PACKET\_SERVICE | MBIM\_PACKET\_SERVICE\_INFO\_V2 |
| MBIM\_CID\_SIGNAL\_STATE | MBIM\_SIGNAL\_STATE\_INFO\_V2 |

## MBIM\_CID\_VERSION

### DESCRIPTION

This is an optional command for exchange of the MBIM version information between the host and the device. If the device requires the backward compatibility with the older MBIM versions, it must support this command.

The host will send query if supported by the device. The query holds the host’s native MBIM Release number and MBIM Extensions Release number the host is currently supporting.

On the device side, the device adjusts its announced MBIM Release number and Extensions Release number based on the rules defined in section 4.2.1, and send them in the response to the host.

The command is defined under the Service Name = **Basic Connect Extensions**

|  |  |  |
| --- | --- | --- |
| **CID** | **Command code** | **UUID** |
| MBIM\_CID\_VERSION | 15 | 3d01dcc5-fef5-4d05-0d3abef7058e9aaf |

### PARAMETERS

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | NA | MBIM\_VERSION\_INFO | NA |
| Response | NA | MBIM\_VERSION\_INFO | NA |

### SET

N/A

### QUERY

Table 2.2‑1: MBIM\_VERSION\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 2 | bcdMBIMVersion | UINT16 | MBIM Release number of the sender in BCD, with implied decimal point between bits 7 and 8. 0x0100 == 1.00 == 1.0. This is a little-endian constant, so the bytes will be 0x00, 0x01. |
| 2 | 2 | bcdMBIMExtendedVersion | UINT16 | MBIM extensions Release number of the sender in BCD, with implied decimal point between bits 7 and 8. 0x0100 == 1.00 == 1.0. This is a little-endian constant, so the bytes will be 0x00, 0x01. |

### RESPONSE

See Table 2.2‑1.

### NOTIFICATION

N/A

### STATUS CODES

This CID only uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

## MBIM\_CID\_MS\_DEVICE\_CAPS\_V2

### DESCRIPTION

This CID is same as defined in [2]. The only difference is that there are new data classes defined in MBIM\_DATA\_CLASS table, which allows the device reporting its 5G capabilities. MBIMDataClass5G\_NSA devotes that the device supports 5G Non-Standalone mode defined in [3GPP TS 37.340](http://www.3gpp.org/DynaReport/37-series.htm) and MBIMDataCLass5G\_SA devotes that the device supports 5G Standalone mode defined in [3GPP TS 37.340](http://www.3gpp.org/DynaReport/37-series.htm).

If the device support both new data classes, then both bits shall be set.

### DATA STRUCTURES

Table 2.3‑1: MBIM\_DATA\_CLASS

|  |  |
| --- | --- |
| Types | Mask |
| MBIMDataClassNone | 0h |
| MBIMDataClassGPRS | 1h |
| MBIMDataClassEDGE | 2h |
| MBIMDataClassUMTS | 4h |
| MBIMDataClassHSDPA | 8h |
| MBIMDataClassHSUPA | 10h |
| MBIMDataClassLTE | 20h |
| MBIMDataClass5G\_NSA | 40h |
| MBIMDataClass5G\_SA | 80h |
| Reserved | 100h-8000h |
| MBIMDataClass1XRTT | 10000h |
| MBIMDataClass1XEVDO | 20000h |
| MBIMDataClass1XEVDORevA | 40000h |
| MBIMDataClass1XEVDV | 80000h |
| MBIMDataClass3XRTT | 100000h |
| MBIMDataClass1XEVDORevB | 200000h |
| MBIMDataClassUMB | 400000h |
| Reserved | 800000-40000000h |
| MBIMDataClassCustom | 80000000h |

## MBIM\_CID\_REGISTER\_STATE

### DESCRIPTION

This command is extension for already existing MBIM\_CID\_REGISTER\_STATE CID defined in [1].

This extension adds a new member called *“PreferredDataClasses”* for the response structure.

### PARAMETERS

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM\_SET\_REGISTRATION\_STATE | Empty | NA |
| Response | MBIM\_REGISTRATION\_STATE\_INFO\_V2 | MBIM\_REGISTRATION\_STATE\_INFO\_V2 | MBIM\_REGISTRATION\_STATE\_INFO\_V2 |

### SET

The information is same as described in [1].

### QUERY

The information shall be null and the InformationBufferLength shall be zero.

### RESPONSE

The following structure shall be used in the InformationBuffer. Compared with the structure MBIM\_CID\_REGISTRATION\_STATE\_INFO defined in 10.5.10.6 of [1], the following structure has a new *PreferredDataClasses* field. Unless stated here, the field descriptions in table 10-55 of [1] apply here.

Table 2.4‑1: MBIM\_ REGISTRATION\_STATE\_INFO\_V2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Offset | Size | | Field | Type | Description |
| 0 | | 4 | NwError | UINT32 | A network-specific error code that comes from mobile operator network related to registration. Possible error codes include, but not limited to, those  documented in the Cause values in the appendixes of the 3GPP TS 24.008 Specification. |
| 4 | | 4 | RegisterState | MBIM\_REGISTER\_STATE | See Table 10-46 in [1]. |
| 8 | | 4 | RegisterMode | MBIM\_REGISTER\_MODE | See Table 10-47 in [1]. |
| 12 | | 4 | AvailableDataClass | UINT32 | A bitmap of the values in Table 4.3.1.2-1: MBIM\_DATA\_CLASS that represent the supported data classes in the registered network, for the cell the device is registered in.  The value is set to MBIMDataClassNone if the RegisterState is not MBIMRegisterStateHome, MBIMRegisterStateRoaming or MBIMRegisterStatePartner. |
| 16 | | 4 | CurrentCellularClass | MBIM\_CELLULAR\_CLASS | Indicates the current cellular class in use for a multi-mode function. See Table 10-8: MBIM\_CELLULAR\_CLASS [1]. For a single-mode function this is the same as the cellular class reported in MBIM\_CID\_DEVICE\_CAPS. For multi-mode functions, a transition from CDMA to GSM or vice versa is indicated with an updated CurrentCellularClass |
| 20 | | 4 | ProviderIdOffset | OFFSET | Offset in bytes, calculated from the beginning of this structure, to a numeric (0-9) string ProviderId that represents the network provider identity.  For GSM-based networks, this string is a concatenation of a three-digit Mobile Country Code (MCC) and a two or three-digit Mobile Network Code (MNC). GSM-based carriers may have more than one MNC, and hence more than one ProviderId .  CDC MBIM Subclass  Mobile Broadband Interface Model 118 May 1, 2013      For CDMA-based networks, this string is a five-digit System ID (SID). Generally, a CDMA-based carrier has more than one SID. Typically, a carrier has one SID for each market, which is usually divided geographically within a nation by regulations, such as Metropolitan Statistical Areas (MSA) in the United States of America. Devices of CDMA-based devices must specify MBIM\_CDMA\_DEFAULT\_PROVIDER\_ ID if this information is not available.  When processing a query request, and the registration state is in automatic register mode, this member contains the provider ID that the device is currently associated with (if applicable). When the registration state is in manual register mode, this member contains the provider ID that the device is requested to register with (even if the provider is unavailable).  When processing a set request and the registration state is in manual mode, this contains the provider ID selected by the Host for the device to register with. When the registration state is in automatic register mode, this parameter is ignored.  CDMA 1xRTT providers must be set to MBIM\_CDMA\_DEFAULT\_PROVIDER\_ ID if the provider ID is not available. |
| 24 | | 4 | ProviderIdSize | SIZE (0 .. 12) Size used for ProviderId | SIZE (0 .. 12) Size used for ProviderId |
| 28 | | 4 | ProviderNameOffset | OFFSET | Offset in bytes, calculated from the beginning of this structure, to a string ProviderName that represents the network provider's name. This member is limited to, at most, MBIM\_PROVIDERNAME\_LEN characters.  For GSM-based networks, if the Preferred Presentation of Country Initials and Mobile Network Name (PPCI&N) is longer than twenty characters, the device should abbreviate the network name.  This member is ignored when the Host sets the preferred provider list.  Devices should specify a NULL string for devices that do not have this information. |
| 32 | | 4 | ProviderNameSize | SIZE (0 .. 40) | Size used for ProviderName |
| 36 | | 4 | RoamingTextOffset | OFFSET | Offset in bytes, calculated from the beginning of this structure, to a string RoamingText to inform the user that the device is roaming. This member is limited to at most 63 characters.  This text should provide additional information to the user when the registration state is either MBIMRegisterStatePartner or MBIMRegisterStateRoaming. This member is optional. |
| 40 | | 4 | RoamingTextSize | SIZE (0 .. 126) | Size used for RoamingText |
| 44 | | 4 | RegistrationFlag | MBIM\_REGISTRATION\_F LAGS | Flag set per Table 10-48: MBIM\_REGISTRATION\_FLAGS. |
| 48 | | 4 | PreferredDataClasses | UINT32 | A bitmap of the values inTable 2.3‑1: MBIM\_DATA\_CLASS that represents the enabled data classes in device. The device can only operate using data classes enabled. |
| <dynamic> | | 4 | DataBuffer | DATABUFFER | ProviderId  ProviderName  RoamingText |

### NOTIFICATION

See Table 2.4‑1.

### STATUS CODES

This CID only uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

## MBIM\_CID\_PACKET\_SERVICE

### DESCRIPTION

This command is extension for already existing MBIM\_CID\_PACKET\_SERVICE CID defined in [1].

This extension adds a new member called *“FrequencyRange”* for the response structure and renames the “HighestAvailableDataClass” member for “*CurrentDataClass*” for more clearer purpose.

The *CurrentDataClass* indicates the Radio Access Technology the device is currently registered. It holds a single value from MBIMDataClass table 4.2.1.2-1.

The *FrequencyRange* indicates the frequency range the device is currently using. This is valid only if the *CurrentDataClasses* field indicates MBIMDataCLass5G\_NSA or MBIMDataClass5G\_SA bit is set.

### PARAMETERS

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM\_SET\_PACKET\_SERVICE | Empty | NA |
| Response | MBIM\_PACKET\_SERVICE\_INFO\_V2 | MBIM\_PACKET\_SERVICE\_INFO\_V2 | MBIM\_PACKET\_SERVICE\_INFO\_V2 |

### DATA STRUCTURES

Table 2.5‑1: MBIM\_FREQUENCY\_RANGE

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMFrequencyRangeUnknown | 0 | If system type is not 5G |
| MBIMFrequencyRange1 | 1 | FR1 in 3GPP TS 38.101-1 (Sub-6G) |
| MBIMFrequencyRange2 | 2 | FR2 in 3GPP TS 38.101-2 (mmWave) |
| MBIMFrequencyRange1AndRange2 | 3 | If both FR1 and FR2 carriers are connected |

### SET

The information is same as described in [1].

### QUERY

The information shall be null and the InformationBufferLength shall be zero.

### RESPONSE

The following structure shall be used in the InformationBuffer. Compared with the structure MBIM\_CID\_PACKET\_SERVICE\_INFO defined in 10.5.10.6 of [1], the following structure has *CurrentDataClass* and *FrequencyRange* new fields. Unless stated here, the field descriptions in table 10-55 of [1] apply here.

Table 2.5‑2: MBIM\_PACKET\_SERVICE\_INFO\_V2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | NwError | UINT32 | A network-specific error code that comes from mobile operator network related to packet service. Possible error codes include, but not limited to, those  documented in the Cause values in the appendixes of the 3GPP TS 24.008 Specification. |
| 4 | 4 | PacketServiceState | MBIM\_PACKET\_SERVICE\_STATE | See Table 10-53 in [1]. |
| 8 | 4 | CurrentDataClass | MBIM\_DATA\_CLASS | The current data class in the current cell, specified according to 4.3.1.2-1: MBIM\_DATA\_CLASS .  Functions must set this member to MBIMDataClassNone if the function is not in packet service state attached.  Except for HSPA (i.e., HSUPA and HSDPA) and 5G DC, the function sets this member to a single MBIM\_DATA\_CLASS value. For HSPA data services, functions specify a bitwise OR of MBIMDataClassHSDPA and MBIMDataClassHSUPA. For cells that support HSDPA but not HSUPA, only HSDPA is indicated (implying UMTS data class for uplink data).  Whenever the current data class changes, functions shall send a notification indicating the new value of CurrentDataClass. |
| 12 | 8 | UplinkSpeed | UINT64 | Contains the uplink bit rate, in bits per second. |
| 20 | 8 | DownlinkSpeed | UINT64 | Contains the downlink bit rate, in bits per second |
| 28 | 4 | FrequencyRange | MBIM\_FREQUENCY\_RANGE | A bitmask of the values in Table 4.3.2.3-1: MBIM\_FREQUENCY\_RANGE representing the frequency ranges which the device is currently using. This is only valid if the CurrentDataClass is either MBIMDataClass5G\_NSA or MBIMDataClass5G\_SA. |

### STATUS CODES

This CID only uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

## MBIM\_CID\_SIGNAL\_STATE

### DESCRIPTION

This chapter specifies a new extension for MBIM\_CID\_SIGNAL\_STATE introducing RSRP and SNR for signal state criterial. The new extension is only valid if the device indicates support of MBIM Extended version 2.0. This extension is mandatory if modem supports MBIMDataClass5G\_(N)SA data classes.

The RSRP and the SNR fields are only valid if the correspondence SystemType is either MBIMDataClassLTE or MBIMDataClass5G\_(N)SA. If the modem reports RSRP and/or SNR, then the RSSI field shall be set to value 99.

If the correspondence SystemType is MBIMDataClass5G\_(N)SA, the RSRP field is mandatory and SNR field is optional. If the correspondence SystemType is MBIMDataClassLTE, the RSRP/SNR fields are optional and RSSI field can be used instead. In this case, RSRP/SNR field can be omitted by setting value zero for both RsrpSnrOffset and RsrpSnrSize members.

The RSRP and SNR mapping tables are based on 3GPP TS 36.336 specification [3].

### PARAMETERS

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM\_SET\_SIGNAL\_STATE | Empty | NA |
| Response | MBIM\_SIGNAL\_STATE\_INFO\_V2 | MBIM\_SIGNAL\_STATE\_INFO\_V2 | MBIM\_SIGNAL\_STATE\_INFO\_V2 |

### SET

The information is same as described in [1].

### QUERY

The *InformationBuffer* shall be null and *InformationBufferLength* shall be zero.

### RESPONSE

The following structure shall be used in the *InformationBuffer*. Compared with the structure MBIM\_SIGNAL\_STATE\_INFO defined in section 10.5.11.5 of [1], the following structure has new RsrpSnrOffset, RsrpSnrSize and DataBuffer fields.

Table 2.6‑1: MBIM\_SIGNAL\_STATE\_INFO\_V2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | Rssi | UINT32 | Same as Table 10.58 in [1] |
| 4 | 4 | ErrorRate | UINT32 | Same as Table 10.58 in [1] |
| 8 | 4 | SignalStrengthInterval | UINT32 | The reporting interval, in seconds |
| 12 | 4 | RssiThreshold | UINT32 | The difference in RSSI coded values that trigger a report. Use 0xffffffff for, do not care. |
| 16 | 4 | ErrorRateThreshold | UINT32 | The difference in ErrorRate coded values that trigger a report. Use 0xffffffff for, do not care. |
| 20 | 4 | RsrpSnrOffset | OFFSET | Offset in bytes, calculated from the beginning of this structure, to the buffer containing RSRP and SNR signaling info. This member can be NULL when no RSRP and SNR signaling info is available |
| 24 | 4 | RsrpSnrSize | SIZE | Total size in bytes of the buffer containing the RSRP and SNR signaling info in the format of **MBIM\_RSRP\_SNR\_INFO**. |
|  | 4 | DataBuffer | DATABUFFER | MBIM\_RSRP\_SNR |

Table 2.6‑2: MBIM\_RSRP\_SNR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | ElementCount | UINT32 | Count of RSRP\_SNR entries following this element |
| 4 | 4 | DataBuffer | DATABUFFER | Array of RSRP\_SNR records, each specified as MBIM\_RSRP\_SNR\_INFO |

Table 2.6‑3: MBIM\_RSRP\_SNR\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | RSRP | UINT32 | |  |  | | --- | --- | | RSRP value in dBm | Coded value (min=0, max = 126) | | less than -156 | 0 | | less than -155 | 1 | | … | … | | less than -138 | 18 | | … | … | | less than -45 | 111 | | … | … | | less than -31 | 125 | | -31 or greater | 126 | | Unknown or undetectable | 127 | |
| 4 | 4 | SNR | UINT32 | |  |  | | --- | --- | | SNR value in dB | Coded value (min=0, max=127) | | less than -23 | 0 | | less than -22.5 | 1 | | less than -22 | 2 | | less than -21.5 | 3 | | … | … | | less than 39.5 | 125 | | less than 40 | 126 | | 40 or greater | 127 | | Unknown or undetectable | 128 | |
| 8 | 4 | RSRPTreshold | UINT32 | Defines the threshold between old (cached) RSRP value and newly calculated RSRP value. If the absolute difference is larger than the threshold value, the device shall trigger unsolicited event. The unit is 1dBm. If set to zero, use default behavior in the device function. If set to 0xffffffff, don’t use this to trigger the event. If the given threshold value is not supported by the device, it shall return the max threshold value it supports. |
| 12 | 4 | SNRThreshold | UINT32 | Defines the threshold between old (cached) SNR value and newly calculated SNR value. If the absolute difference is larger than the threshold value, the device shall trigger unsolicited event. The unit is 1dB. If set to zero, use default behavior in the device function. If set to 0xffffffff, don’t use this to trigger the event. If the given threshold value is not supported by the device, it shall return the max threshold value it supports. |
| 16 | 4 | SystemType | MBIM\_DATA\_CLASS | Indicates the system type for which signal state information is valid, a bitmask of one type as defined in the 4.3.1.2-1: MBIM\_DATA\_CLASS |

### NOTIFICATION

See Section 2.6.5.

### STATUS CODES

This CID only uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

# MBIM Interface Extensions for 5G NGC – Phase 1

This section defines the MBIM extensions for first phase of features for 5G system with the next generation core network (NGC, aka 5GC), including deployment options 2, 4, 7 and 5. These extensions build on top of the MBIM extensions for 5G NSA defined in Section 2. This phase of 5G NGC features are those intended for 5G SA commercialization in 2021H1 with Windows Fe release. It enables MBB functionality parity for Windows MBIM MBB devices on 5G SA networks vs 5G NSA, including registration and basic PDU sessions on default eMBB network slice. On the other hand, this phase excludes any specific support for more advanced 5G NGC features such as:

* Non-eMBB network slice
* multiple concurrent network slices
* End to end URSP & ANDSP processing and support
* VoNR or e911 in 5G NGC
* application/cloud level secondary authentication
* Non-3GPP access
* unstructured or non-IP PDU sessions
* advanced metering/cost management support for 5G links
* URLLC or massive IoT support.

In this section, the terms “modem” and “device” refer to the MBIM device that an MBIM host communicates with.

## Versioning Scheme

A new MBIM Extensions Release number 3.0 is introduced for the changes specified in Section 3. Those changes include breaking changes from the existing MBIM Extensions Release numbers 1.0 and 2.0.

As in Section 2.1, the term “MBIMEx version” refers to the MBIM Extensions Release number of this document.

MBIMEx version 3.0 is under existing MBIM Release number 1.0, like MBIMEx version 1.0 [1] and MBIMEx version 2.0 defined in Section 2.

Unless explicitly documented otherwise, all procedures for versioning defined in Section 2.1 apply to both host and device. A host and a device shall follow those procedures to learn one another’s version capabilities and to subsequently agree to an MBIMEx version to govern the MBIM interface for the duration of the device being enumerated to the host.

A device capable of MBIMEx 3.0 and with intention to work with older hosts should initially advertise MBIMEx version 1.0 in MBIM extended functional descriptor. If the host sends MBIM\_CID\_VERSION and the host has an MBIMEx version no less than 3.0, the device should, in the MBIM\_CID\_VERSION response, indicates MBIMEx 3.0. At that point, the host and the device agree to use MBIMEx 3.0.

Unless explicitly mentioned and modified, all unmentioned payloads, CIDs and procedures in MBIMEx versions 2.0 and 1.0 carry over and stay unchanged in MBIMEx version 3.0.

The new CIDs introduced in MBIMEx 3.0 are listed in Table 3.1‑1.

Table 3.1‑1: Command Codes and Belonging Service of New CIDs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CID** | **Command code** | **Service Name** | **Maximum Allowed Time for MBIM Device to Respond (in seconds)** | | **Is the CID mandatory?** |
| **Set** | **Query** |
| MBIM\_CID\_MS\_MODEM\_CONFIG | 16 | Basic Connect Extensions | N/A | 8 | yes |
| MBIM\_CID\_MS\_REGISTRATION\_PARAMS | 17 | Basic Connect Extensions | 8 | 8 | yes |
| MBIM\_CID\_MS\_NETWORK\_PARAMS | 18 | Basic Connect Extensions | N/A | 58 | no |
| MBIM\_CID\_MS\_WAKE\_REASON | 19 | Basic Connect Extensions | N/A | 10 | no |

The modified existing CIDs are listed in Table 3.1‑2. These CIDs retain their existing command code and associated service.

Table 3.1‑2: Modified Existing CIDs

|  |  |  |  |
| --- | --- | --- | --- |
| **CID** | **Maximum Allowed Time for MBIM Device to Respond (in seconds)** | | **Is the CID mandatory or optional?** |
| **Set** | **Query** |
| MBIM\_CID\_MS\_DEVICE\_CAP\_V2 | N/A | 8 | mandatory |
| MBIM\_CID\_SUBSCRIBER\_READY\_STATUS | N/A | 8 | No change from previous MBIMEx version |
| MBIM\_CID\_PACKET\_SERVICE | 58 | 8 | No change from previous MBIMEx version |
| MBIM\_CID\_CONNECT | 198 | 8 | No change from previous MBIMEx version |
| MBIM\_CID\_IP\_PACKET\_FILTERS | 8 | 8 | No change from previous MBIMEx version |
| MBIM\_CID\_BASE\_STATIONS\_INFO | N/A | 8 | No change from previous MBIMEx version |
| MBIM\_CID\_MS\_LTE\_ATTACH\_STATUS | N/A | 8 | No change from previous MBIMEx version |

For each new or modified CID in MBIMEx 3.0, if requests for set or query are required, MBIM devices shall process the requests in timely manner, complete the processing and send back responses. Each new or modified CID has a maximum allowed time for MBIM devices to respond to a query request (if query request is required), and one for a set request (if set request is required). The time limits are listed above in the tables along with each new or modified CID. An MBIM device shall send a response within the maximum allowed response time. Late response is invalid and will be discarded by the host.

The optional CIDs added to MBIMEx 1.0 for eSIM support are mandatory in MBIMEx 3.0, so that eSIM related ready state values must be recognized by hosts and devices in MBIMEx 3.0. Note that this does not imply in any way whether an eSIM is present in a particular device.

## MBIM TLV-Typed Information Element

This section defines MBIM\_TLV\_IE, a variable-sized data structure that is capable of exchanging a wide range of information between an MBIM host and an MBIM device.

CID payloads for requests, responses, and/or notifications may contain zero or more unnamed and optional Information Elements (IE) encoded as MBIM\_TLV\_IE. These optional elements, if present, will appear at the end of the payload. CID payloads can also use MBIM\_TLV\_IE for named and/or mandatory fields.

MBIM\_TLV\_IE contains a Type field for identifying the type of the data present in the structure, a DataLength field that specifies the size of the data contained in the Data field, a PaddingLength field for the size of padding contained in the the Padding field, a Data field to carry data, and a Padding field for the padding.

The padding is to ensure that the entire size of the structure is a multiple of the size of a DWORD. That is, the entire size of the structure, including Type, Length, Data and Padding, shall be a multiple of 4 octets. The size of Padding is 0, 1, 2, or 3 octet(s). As the entire structure size is a multiple of the size of a DWORD, the end of the structure will align to a DWORD boundary if the structure starts at a DWORD boundary.

MBIM\_TLV\_IE is defined in Table 3.2‑1. The Type field in a must contain one of the type values defined in Section 3.2.2.

Table 3.2‑1: MBIM\_TLV\_IE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octet | Size in octet | Field | Type | Description |
| 0 | 2 | Type | UINT16 | A two-octet IE type value identifying the information element that follows. See Section 3.2.2 for possible values. |
| 2 | 1 | Reserved | UINT8 | Unused and reserved byte. Must set to zero |
| 3 | 1 | PaddingLength | UINT8 | Length of the Padding field in octets. Must be 0, 1, 2, or 3. |
| 4 | 4 | DataLength | UINT32 | The length of the Data field in octets. |
| 8 | Variable | Data | Data buffer in octets | The data of the Information Element. The format and interpretation of the data in this field is specific to the type specified by the Type field. |
| Variable | Variable. Maximum is 3 | Padding | Padding octets | The padding is to ensure that the entire structure is DWORD aligned. That is, the entire size, including Type, Length, Data and Padding, is a multiple of 4 octets. The value of padding bytes must be set to 0. |

### General Rules of Processing Optional TLV-typed IEs

Upon receiving a CID whose payload may contain optional, unnamed IE’s in in MBIM\_TLV\_IE format, the receiver shall follow the general rules below to parse and process optional IEs in the payload in MBIM\_TLV\_IE format:

1. Named fields always come before unnamed IE’s in MBIM\_TLV\_IE format in the payload. Named fields should be parsed first. If there is remaining data after named fields as determined by *InformationBufferLength* minus the total length of the named fields, continue as below.
2. The receiver shall attempt to parse an optional information element (IE) in MBIM\_TLV\_IE structure in the remaining data. There is a valid MBIM\_TLV\_IE data structure in the remaining data if both of the following are true:
   1. The remaining data length is no less than 8 octets for a possible MBIM\_TLV\_IE structure.
   2. The remaining data length, minus 8 octets for the fixed part of MBIM\_TLV\_IE, is no less than potential data and padding required by the PaddingLength and DataLength fields in MBIM\_TLV\_IE.
3. If a valid MBIM\_TLV\_IE data structure is not found in the remaining data, the entire payload is considered invalid and the parsing of the payload is complete.
4. If a valid MBIM\_TLV\_IE data structure is found, the receiver checks the Type of the MBIM\_TLV\_IE data.
   1. If the IE is recognized and supported, the receiver shall process it and act according to the information in the IE.
   2. If the IE is not recognized or not supported, the IE is silently discarded.
5. After processing the found IE MBIM\_TLV\_IE data structure, if there is remaining data as determined by *InformationBufferLength* minus the total length of the named fields and all processed unnamed fields, go back to Step 2) above and continue.

The rules defined in this section apply by default, unless explicitly specified otherwise for individual requests, responses, and/or notifications.

### Types for MBIM TLV-typed Information Elements

All type values for MBIM\_TLV\_IE are listed in the Table 3.2‑2. The description column in the table provides a general description for each of the types. It also defines, or points to the definition of, the corresponding structure of the data contained within the Data field.

The valid range of the type values is 1 – 0xFFF0. The rest of values are reserved.

Table 3.2‑2: Type values for MBIM\_TLV\_IE

|  |  |  |
| --- | --- | --- |
| TLV Type value | TLV Type Name | Description |
| 0 | Reserved | Reserved/invalid TLV type value |
| 1 | MBIM\_TLV\_TYPE\_UE\_POLICIES | UE policies. The structure of the Data field, and its usage, are specified in Section 3.8. |
| 2 | MBIM\_TLV\_TYPE\_SINGLE\_NSSAI | Single NSSAI. The Data field in this type of TLV IE contains one S\_NSSAI in MBIM\_MS\_SNSSAI\_INFO format (Table 3.2‑4). The size of MBIM\_MS\_SNSSAI\_INFO data structure is variable depending on its SnssaiLength field. The size of the Data field derived from the Length field in MBIM\_TLV\_IE must agree to the size implied by the SnssaiLength field in MBIM\_MS\_SNSSAI\_INFO. In the event they do not agree, the TLV is invalid and the CID containing the TLV is considered to have invalid parameter. |
| 3 | MBIM\_TLV\_TYPE\_ALLOWED\_NSSAI | Allowed NSSAI. The Data field in this type of TLV IE contains one or more S-NSSAIs, each in the MBIM\_MS\_SNSSAI\_INFO format (Table 3.2‑4). The size specified in the MBIM\_TLV\_IE Length field must reflect the combined S-NSSAI lengths. A S-NSSAI’s length equals the SnssaiLength value plus 1 octet for the SnssaiLength field itself. In the event the lengths do not correspond, the TLV is invalid and the CID containing the TLV is considered to have invalid parameters. |
| 4 | MBIM\_TLV\_TYPE\_CFG\_NSSAI | Configured NSSAI. The Data field in this type of TLV IE contains one or more S-NSSAIs, each in the MBIM\_MS\_SNSSAI\_INFO format (Table 3.2‑4). The size specified in the MBIM\_TLV\_IE Length field must reflect the combined S-NSSAI lengths. A S-NSSAI’s length equals the SnssaiLength value plus 1 octet for the SnssaiLength field itself. In the event the lengths do not correspond, the TLV is invalid and the CID containing the TLV is considered to have invalid parameters. |
| 5 | MBIM\_TLV\_TYPE\_DFLT\_CFG\_NSSAI | Default Configured NSSAI. The Data field in this type of TLV IE contains one or more S-NSSAIs, each in the MBIM\_MS\_SNSSAI\_INFO format (Table 3.2‑4). The size specified in the MBIM\_TLV\_IE Length field must reflect the combined S-NSSAI lengths. A S-NSSAI’s length equals the SnssaiLength value plus 1 octet for the SnssaiLength field itself. In the event the lengths do not correspond, the TLV is invalid and the CID containing the TLV is considered to have invalid parameters. |
| 6 | MBIM\_TLV\_TYPE\_PRECFG\_DFLT\_CFG\_NSSAI | Preconfigured default Configured NSSAI. The Data field in this type of TLV IE contains one or two preferred NSSAI List per access type defined in the MBIM\_MS\_PRE\_DFLT\_NSSAI\_INFO format (Table 3.2‑5). The size specified in the MBIM\_TLV\_IE Length field must reflect the total of combined access type and PreferredNSSAI lengths per each different access type. |
| 7 | MBIM\_TLV\_TYPE\_REJ\_NSSAI | Rejected NSSAI. The Data field in this type of TLV IE contains one or more Rejected S-NSSAIs, each in the MBIM\_MS\_REJ\_SNSSAI\_INFO format (Table 3.8‑3). The size specified in the MBIM\_TLV\_IE Length field must reflect the combined Rejected S-NSSAI lengths. A Rejected S-NSSAI’s length equals the SnssaiLength value plus 2 octets for the SnssaiLength and Cause fields. In the event the lengths do not correspond, the TLV is invalid and the CID containing the TLV is considered to have invalid parameters. |
| 8 | MBIM\_TLV\_TYPE\_LADN | Local Area Data Network (LADN). The Data field in this type of TLV IE contains one or more LADNs, each in the MBIM\_MS\_LADN format (Table 3.8‑4). The size specified in the MBIM\_TLV\_IE Length field must reflect the length of the combined LADNs. The length of each LADN is determined by the length of DNN value field and the length of tracking area identity list field. |
| 9 | MBIM\_TLV\_TYPE\_TAI | Tracking Area Identity (TAI). The Data field in this type of TLV IE contains one or more TAI lists, each in the MBIM\_MS\_TAI\_LIST\_INFO format (Table 3.2‑3). The size specified in the MBIM\_TLV\_IE Length field must reflect the combined length of all the tracking area identity lists. The length of each partial tracking area identity list is determined from the 'type of list' field and the 'number of elements' fields. |
| 10 | MBIM\_TLV\_TYPE\_WCHAR\_STR | WCHAR string. The Data field contains a WCHAR string with no null-terminator. Each character occupies two consecutive octets in the Data field. The DataLength must be a multiple of 2. |
| 11 | MBIM\_TLV\_TYPE\_UINT16\_TBL | Array of 1 or more UINT16 entries. The Data field contains an array of UINT16 entries. Each entry occupies two consecutive octets in the Data field. The DataLength must be a multiple of 2. |
| 12 | MBIM\_TLV\_TYPE\_EAP\_PACKET | Extensible Authentication Protocol packet. The Data field in this type of TLV IE contains EAP packet from Authenticator (network) or Peer (MBIM host) |
| 13 | MBIM\_TLV\_TYPE\_PCO | Protocol Configuration Option. The Data field in this type of TLV IE contains one or more PCO options specified in TS 24.008 |
| 14 | MBIM\_TLV\_TYPE\_ROUTE\_SELECTION\_DESCRIPTORS | The Data field in this type of TLV IE contains one or more components each in the MBIM\_MS\_ROUTE\_SELECTION\_DESCRIPTOR format (Table 3.2‑8). The size specified in the MBIM\_TLV\_IE Length field must reflect the combined length of all the Route Selection Descriptor Info structures which are present. |
| 15 | MBIM\_TLV\_TYPE\_TRAFFIC\_PARAMETERS | The data field in this type of TLV IE contains a MBIM\_MS\_TRAFFIC\_PARAMETERS record. |
| 16 | MBIM\_TLV\_TYPE\_WAKE\_COMMAND | The data field in this type of TLV IE contains a MBIM\_MS\_WAKE\_COMMAND |
| 17 | MBIM\_TLV\_TYPE\_WAKE\_PACKET | The data field in this type of TLV IE contains a MBIM\_MS\_WAKE\_PACKET |

### Data Structures for Commonly Used TLV Types

The sections define the structures and encoding of the Data field for commonly used types of TLV IEs.

The following table defines the tracking area identity list data format for MBIM\_TLV\_TYPE\_TAI.

A Tracking Area Identity (TAI) consists of a PLMN + tracking area code (TAC). The MBIM\_MS\_TAI\_LIST\_INFO structure includes one or more partial TAI lists. The format of each list depends on the specified type.

Table 3.2‑3: MBIM\_MS\_TAI\_LIST\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 1 | ListType | UINT8 | Partial tracking area identity list type 0 - List of TACs belonging to one PLMN, with non-consecutive TAC values 1 - List of TACs belonging to one PLMN, with consecutive TAC values 2 - List of TAIs belonging to different PLMNs |
| 1 | Variable | ListData | DataBuf | Partial tracking area identity list data. Format depends on the specified list type.  1.MBIM\_MS\_TAI\_LIST\_SINGLE\_PLMN (Table 4.1‑4) used single PLMN list (Types 0 & 1) 2.MBIM\_MS\_TAI\_LIST\_MULTI\_PLMNS (Table 4.1‑5) used for List of TAIs belonging to different PLMNs (Type 2) |

The following defines the S-NSSAI data format for the following NSSAI TLVs:

* MBIM\_TLV\_TYPE\_ALLOWED\_NSSAI
* MBIM\_TLV\_TYPE\_CFG\_NSSAI
* MBIM\_TLV\_TYPE\_DFLT\_CFG\_NSSAI

Each S-NSSAI value consists of one S-NSSAI and optionally one mapped configured S-NSSAI.

Table 3.2‑4: MBIM\_MS\_SNSSAI\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 1 | SnssaiLength | UINT8 | Length is expressed per 3GPP TS 24. 501 section 9.11.2.8.  The length of the included S-NSSAI contents in octets.  Depending on the value of the length field the following S-NSSAI contents are included: 1 - SST 2 - SST and mapped HPLMN SST 4 - SST and SD 5 - SST, SD and mapped HPLMN SST 8 - SST, SD, mapped HPLMN SST and mapped HPLMN SD All other values are reserved |
| 1 | 1 | SliceServiceType | UINT8 | Slice/service type (SST) The SST field may have standardized and non-standardized values. Values 0 to 127 belong to the standardized SST range. Values 128 to 255 belong to the Operator-specific range. Standardized values: 1 - Enhanced Mobile Broadband 2 - Ultra-reliable low latency communications. 3 - Massive IoT |
| variable | 3 | Sd | UINT8 | Slice differentiator (SD). Optional information that complements the Slice/Service type(s) to differentiate amongst multiple Network Slices.  The SD field has a reserved value "no SD value associated with the SST" defined as hexadecimal FFFFFF |
| variable | 1 | MappedSst | UINT8 | Mapped HPLMN SST |
| variable | 3 | MappedSd | UINT8 | Mapped HPLMN SD |

Table 3.2‑5: MBIM\_MS\_PRE\_DFLT\_NSSAI\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 4 | AccessType | MBIM\_MS\_ACCESS\_TYPE | Access type. Mandatory information that specifies the access type of the preconfigured default NSSAI list. Access type can be either 3GPP or Non-3GPP. |
| 4 | Variable | PreferredNSSAI | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_DFLT\_CFG\_NSSAI | Contain variable amount of information for one or more S\_NSSAI in TLV of type MBIM\_TLV\_TYPE\_DFLT\_CFG\_NSSAI. See the S\_NSSAI definition in MBIM\_MS\_SNSSAI\_INFO. When SnssaiLength is 0, no SNSSAI is given. Each MBIM\_NS\_SNSSAI\_INFO can only contain SD and SST values. The device may ignore it if not supported in a RAT type where SNSSAI is not applicable. |

The MBIM\_MS\_ROUTE\_SELECTION\_DESCRIPTOR type is described in Table 3.2‑8. This data type is comprised of a route source (Table 3.2‑6), route purpose details (Table 3.2‑7), and the route selection descriptor details described in [6] TS 24.526 “Policies for 5G System; Stage 3”, Section 5 “Encoding of UE Policies” (Figure 2).

Table 3.2‑6: MBIM\_MS\_ROUTE\_SELECTION\_DESCRIPTOR\_SOURCE

|  |  |  |
| --- | --- | --- |
| Type | Value | Source |
| MBIMRouteSelectionSourceDescriptorDefault | 0h | Default or unspecified |
| MBIMRouteSelectionSourceDescriptorUser | 1h | User |
| MBIMRouteSelectionSourceDescriptorAdmin | 2h | Device administrator |
| MBIMRouteSelectionSourceDescriptorOperator | 3h | Operator configured in HLOS |
| MBIMRouteSelectionSourceDescriptorDevice | 4h | Device |
| MBIMRouteSelectionSourceDescriptorModemOperator | 5h | URSP network update |
| MBIMRouteSelectionSourceDescriptorModemLocal | 6h | URSP pre-configured on device |

Table 3.2‑7: MBIM\_MS\_ROUTE\_SELECTION\_DESCRIPTOR\_PURPOSE

|  |  |
| --- | --- |
| Types | Mask |
| MBIMRouteSelectionDescriptorPurposeDefault | 0h |
| MBIMRouteSelectionDescriptorPurposePurchase | 1h |

Figure 2: Route Selection Descriptor Contents

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |  |
| Length of route selection descriptor | | | | | | | | octet 1  octet 2 |
| Precedence value of route selection descriptor | | | | | | | | octet 3 |
| Length of route selection descriptor contents | | | | | | | | octet 4 |
|  |
| octet 5 |
| Route selection descriptor contents | | | | | | | | octet 6 |
|  |
| octet x |

Table 3.2‑8: MBIM\_MS\_ROUTE\_SELECTION\_DESCRIPTOR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 4 | Source | MBIM\_MS\_ROUTE\_SELECTION\_DESCRIPTOR\_SOURCE | Source which provided this route, see Table 3.2‑6. |
| 4 | 4 | Purpose | MBIM\_MS\_ROUTE\_SELECTION\_DESCRIPTOR\_PURPOSE | Routing purposes applicable to this descriptor, see Table 3.2‑7. |
| 8 | var | RouteSelectionDescriptor | Encoded 5G URSP Route Selection Descriptor | Route selection descriptor as defined in [6] TS 24.526 “Policies for 5G System; Stage 3”, Section 5 “Encoding of UE Policies” and illustrated in Figure 2. |

Figure 3: Traffic Descriptor Length and Value

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |  |
| Length of traffic descriptor | | | | | | | | octet 1  octet 2 |
| Traffic descriptor | | | | | | | | octet 3 |
|  |
| octet x |

Table 3.2‑9: MBIM\_MS\_TRAFFIC\_PARAMETERS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 2 | TrafficDescriptorLength | UINT16 | Encoded as per “Traffic Descriptors” in 3GPP [6] TS24.526 “Policies for 5G System; Stage 3,” Section 5 “Encoding of UE Policies” and shown in Figure 3. This field contains the “Length of traffic descriptor.” |
| var | var | TrafficDescriptor | Encoded 5G Traffic Descriptor | Encoded as per “Traffic Descriptors” in 3GPP [6] TS24.526 “Policies for 5G System; Stage 3,” Section 5 “Encoding of UE Policies” and shown in Figure 3. This field contains the “Traffic descriptor”. At most one traffic component will exist per traffic component type. This field will not be present if TrafficDescriptorLength is 0. |

Table 3.2‑10: MBIM\_MS\_WAKE\_COMMAND

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 16 | DeviceServiceId | UUID | A 16-byte UUID that identifies the device service the following CID value applies. |
| 16 | 4 | CID | UINT32 | Specifies the CID that identifies the wake command |
| 20 | 4 | PayloadOffset | UINT32 | Offset in bytes, calculated from the beginning (offset 0) of this MBIM\_MS\_WAKE\_COMMAND structure, to the wake payload. |
| 24 | 4 | PayloadSize | UINT32 | Specifies the length, in units of bytes, of the wake payload that follows this structure. |
| 28 | Var | DataBuffer | DATABUFFER | Wake payload. Right now, only when the wake CID is MBIM\_CID\_CONNECT, the payload is 4 bytes long to differentiate activate and deactivate. And its value must be 1 = activate or 0 = deactivate.  All payload for other wake CID is ignored |

Table 3.2‑11: MBIM\_MS\_WAKE\_PACKET

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 4 | FilterId | UINT32 | A FilterId specifies a host-provided value that identifies an IP packet filter set by host to modem via MBIM\_CID\_IP\_FILTERS. Modem wakes up because this IP packet filter matched the incoming packet. |
| 4 | 4 | OriginalPacketSize | UINT32 | Specifies the original length, in units of bytes, of the wake packet |
| 8 | 4 | SavedPacketOffset | UINT32 | Offset in bytes, calculated from the beginning (offset 0) of this MBIM\_MS\_WAKE\_PACKET structure, to the saved wake packet data |
| 12 | 4 | SavedPacketSize | UIN32 | Specifies the length, in units of bytes, of the saved wake packet data that follows this structure |
| 16 | Var | DataBuffer | DATABUFFER | Saved wake packet data |

## MBIM\_CID\_MS\_DEVICE\_CAPS\_V2

### DESCRIPTION

The description of this CID in original MBIM 1.0[1] applies, with the enhancements below.

Dataclass names for MBIMDataClass5G\_NSA and MBIMDataClass5G\_SA from types in Section 2.3 are renamed to MBIMDataClass5G with corresponding data subclasses. MBIM device shall report device capabilities as defined in Table 3.3.2-1 and RF bands for WCDMA, LTE and NR. The variable length fields from [2]**.** have been redefined in TLV format at the end of the MBIM\_DEVICE\_CAPS\_INFO\_V3 in Table 3.3.3-1.

### DATA STRUCTURES

Table 3.3‑1: MBIM\_DATA\_CLASS

|  |  |
| --- | --- |
| Types | Mask |
| MBIMDataClassNone | 0h |
| MBIMDataClassGPRS | 1h |
| MBIMDataClassEDGE | 2h |
| MBIMDataClassUMTS | 4h |
| MBIMDataClassHSDPA | 8h |
| MBIMDataClassHSUPA | 10h |
| MBIMDataClassLTE | 20h |
| MBIMDataClass5G | 40h |
| MBIMDataClassUnused | 80h |
| Reserved | 100h-8000h |
| MBIMDataClass1XRTT | 10000h |
| MBIMDataClass1XEVDO | 20000h |
| MBIMDataClass1XEVDORevA | 40000h |
| MBIMDataClass1XEVDV | 80000h |
| MBIMDataClass3XRTT | 100000h |
| MBIMDataClass1XEVDORevB | 200000h |
| MBIMDataClassUMB | 400000h |
| Reserved | 800000-40000000h |
| MBIMDataClassCustom | 80000000h |

The table below defines bitmasks for data sub-classes. A map of these bitmasks represents the radio technology data subclasses that are supported by the function.

Table 3.3‑2: MBIM\_DATA\_SUBCLASS

|  |  |  |
| --- | --- | --- |
| Types | Mask | Description |
| MBIMDataSubClassNone | 0h | No data subclass |
| MBIMDataSubClass5GENDC | 1h | EUTRAN and NR dual connectivity as in 5G Option 3 |
| MBIMDataSubClass5GNR | 2h | Standalone NR as in 5G Option 2 |
| MBIMDataSubClass5GNEDC | 4h | NR and EUTRAN dual connectivity as in 5G Option 4 |
| MBIMDataSubClass5GELTE | 8h | eLTE as in 5G option 5 |
| MBIMDataSubClass5GNGENDC | 10h | Next-gen eLTE and NR dual connectivity as in 5G option 7 |

The data type MBIM\_CTRL\_CAPS in previous MBIMEx version is extended as below

Table 3.3‑3: MBIM\_CTRL\_CAPS

|  |  |  |
| --- | --- | --- |
| Types | Mask | Description |
| MBIMCtrlCapsNone | 0h | The same as in [1] |
| MBIMCtrlCapsRegManual | 1h | The same as in [1] |
| MBIMCtrlCapsHwRadioSwitch | 2h | The same as in [1] |
| MBIMCtrlCapsCdmaMobileIp | 4h | The same as in [1] |
| MBIMCtrlCapsCdmaSimpleIp | 8h | The same as in [1] |
| MBIMCtrlCapsMultiCarrier | 10h | The same as in [1] |
| MBIMCtrlCapsESIM | 20h | Indicates whether the device supports ESIM. |
| MBIMCtrlCapsUEPolicyRouteSelection | 40h | Indicates whether the device supports including the Route Selection Descriptors as part of the MBIM\_TLV\_TYPE\_UE\_POLICIES defined in Section 3.2.2. |
| MBIMCtrlCapsSIMHotSwapCapable | 80h | Indicates whether the device supports SIM hot-swap. |

### PARAMETERS

The table below lists the valid forms of the CID and the corresponding payload.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | NA | Empty | NA |
| Response | NA | MBIM\_DEVICE\_CAPS\_INFO\_V3 | NA |

### RESPONSE

The following structure shall be used in the *InformationBuffer*. Compared with the structure MBIM\_CID\_DEVICE\_CAPS\_V2 defined in [2], the following structure has new fields DeviceCaps, WcdmaBandClass, LteBandClass and NRBandClass. Unless stated here, the field descriptions in Table 10-14 of [1] apply here.

Table 3.3.3‑1: MBIM\_DEVICE\_CAPS\_INFO\_V3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | DeviceType | MBIM\_DEVICE\_TYPE | The same as in MBIM 1.0 in [1] |
| 4 | 4 | CellularClass | MBIM\_CELLUAR\_CLASS | The same as in MBIM 1.0 in [1] |
| 8 | 4 | VoiceClass | MBIM\_VOICE\_CLASS | The same as in MBIM 1.0 in [1] |
| 12 | 4 | SimClass | UINT32 | A bitmap of the masks defined in MBIM\_SIM\_CLASS.For example, if the modem supports both embedded SIM and removable SIM in a potential multiple SIM slot configuration, this field contains  (MBIMSimClassSimLogical |  MBIMSimClassSimRemovable) |
| 16 | 4 | DataClass | MBIM\_DATA\_CLASS | The same as in MBIM 1.0 in [1] |
| 20 | 4 | SmsCaps | MBIM\_SMS\_CAPS | The same as in MBIM 1.0 in [1] |
| 24 | 4 | ControlCaps | MBIM\_CTRL\_CAPS | See the definition of the extended MBIM\_CTRL\_CAPS type. |
| 28 | 8 | DataSubClass | UINT64 | A 64-bit integer containing the map (logical OR) of the bitmasks in MBIM\_DATA\_SUBCLASS that represents the RAT data subclasses that are supported by the device. If a device supports 5G NSA (option 3), the DataClass field must contain the MBIMDataClass5G bit and this field must contain the MBIMDataSubClass5GENDC bit. If a device supports 5G NSA (option 3) and 5G NR standalone, the DataClass field must contain the MBIMDataClass5G bit and this field must contain the MBIMDataSubClass5GENDC and MBIMDataSubClass5GNR bits. |
| 36 | 4 | MaxSessions | UINT32 | The same as in MBIM 1.0 in [1] |
| 40 | 4 | ExecutorIndex | UINT32 | The same as in MBIM 1.0 in [2] |
| 44 | 4 | WcdmaBandClass | UINT32 | The band number defined in 3GPP TS25.101 for FDD. The least significant bit indicates Band number 1 (2100MHz). Zero indicates bands unknown if WCDMA is supported per device capability. |
| Var |  | LteBandClass | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_UINT16\_TBL | LTE bands defined in 3GPP TS 36.101. Each UINT16 entry represents a band number e.g. Band 5 is 5 decimal. Zero-element table in this TLV indicates bands unknown if LTE is supported per device capability. |
| Var |  | NRBandClass | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_UINT16\_TBL | 5G NR bands defined in 3GPP TS 38.101. Each UINT16 entry represents a band number e.g. Band n5 is 5 decimal. Zero-element table in this TLV indicates bands unknown if NR is supported per device capability. |
| Var | 1..11 | CustomDataCLass | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | The same as in MBIM 1.0 in [1] |
| Var | 1..18 | DeviceID | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | The same as in MBIM 1.0 in [1] |
| Var | 1..30 | FirmwareInfo | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | The same as in MBIM 1.0 in [1] |
| Var | 1..30 | HardwareInfo | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | The same as in MBIM 1.0 in [1] |

## MBIM\_CID\_SUBSCRIBER\_READY\_STATUS

### DESCRIPTION

The description of this CID in original MBIM 1.0[1] applies, with enhancement and clarification below.

When a modem reports a ReadyState of MBIMSubscriberReadyStateInitialized in query responses or notifications, all data about the subscriber and/or SIM in the fields of the payload must be valid and final. This includes Telephone Number(s). When an MBIM device has not completed obtaining telephone number(s) for the current SIM, modem shall not report a ReadyState of MBIMSubscriberReadyStateInitialized in query responses or notifications.

Likewise, when a modem reports a ReadyState of MBIMSubscriberReadyStateInitialized in query responses or notifications, the network parameters in Network-set UE policies defined in 3.8 shall be ready for query if the modem is capable of 5G-NGC.

SIM hot-swap is the situation where an user removes the SIM from the system and then insert another SIM into the system, all while the system is running and continue running for entire duration of the user actions. Opposite to SIM hot-swap is SIM cold-swap where SIM is replaced while the system is powered down. SIM hot-swap is treated as an act of SIM removal followed in a short time by an act of SIM insertion.

When modem detects that a SIM is removed, the modem shall immediately send a notification with ReadyState of MBIMSubscriberReadyStateSimNotInserted. When modem detects that a SIM is inserted, the modem shall immediately send a notification with ReadyState of MBIMSubscriberReadyStateNotInitialized, before it starts initializing for the SIM. When the modem finishes initializing the SIM and all information for ReadyState of MBIMSubscriberReadyStateInitialized are obtained, the modem sends a notification with ReadyState of MBIMSubscriberReadyStateInitialized.

For eSIM, disabling an active eSIM profile and leaving the eSIM without an active eSIM profile is equivalent of a physical SIM being removed. Modem shall treat such case as SIM removal, with MBIMSubscriberReadyStateNoEsimProfile being the equivalent ready state to MBIMSubscriberReadyStateSimNotInserted. Likewise, enabling an eSIM profile in an eSIM is equivalent to inserting a physical SIM. Modem shall immediately send a notification with ReadyState of MBIMSubscriberReadyStateNotInitialized before initializing for it, and later sends a notification with ReadyState of MBIMSubscriberReadyStateInitialized after finishing initialization.

When a system boots up with a SIM inserted, the modem should send an unsolicited notification with ReadyState of MBIMSubscriberReadyStateNotInitialized, after it detects the SIM and before it starts initializing for the SIM. However, that notification may not reach the host since the host opens the MBIM communication with the modem at its own pace. The host sends a query when it is ready. Upon receiving a query request, modem shall respond with its then current state.

When SIM is changed during system sleep or hibernation, the modem may not detect the event. Modem is not responsible of detecting SIM swap during system sleep or hibernation. When system resumes from a sleep or hibernation, it is the host’s responsibility to detect whether the SIM is different from the one before the sleep or hibernation without requiring a query response or notification with ReadyState of MBIMSubscriberReadyStateSimNotInserted and/or MBIMSubscriberReadyStateNotInitialized. If the SIM has changed during sleep, the host should cleanup state associated with the previous SIM and use the new SIM as the current SIM.

### DATA STRUCTURES

New bit map masks below are introduced and are used in a new field in query responses and notifications.

Table 3.4‑1: MBIM\_SUBSCRIBER\_READY\_STATUS\_FLAGS

|  |  |  |
| --- | --- | --- |
| Types | Mask | Description |
| MBIMSubscriberReadyStatusFlagNone | 0h | No flag mask |
| MBIMSubscriberReadyStatusFlagESim | 1h | 1 -- Current SIM is an eSIM; 0 – current SIM is not an eSIM. Valid only if eSIM is supported by the modem and when the ReadyState is  MBIMSubscriberReadyStateInitialized or MBIMSubscriberReadyStateNoEsimProfile. |
| MBIMSubscriberReadyStatusFlagSIMRemovabilityKnown | 2h | 1 – whether the SIM is removable is known;  0 -- whether the SIM is removable is unknown |
| MBIMSubscriberReadyStatusFlagSIMRemovable | 4h | 1 -- Current SIM is removable; 0 – current SIM is embedded. Valid only when the ReadyState is  MBIMSubscriberReadyStateInitialized, MBIMSubscriberReadyStateNoEsimProfile, or  MBIMSubscriberReadyStateDeviceLocked.  This bit is valid only if the bit MBIMSubscriberReadyStatusFlagSIMRemovabilityKnown indicates that the SIM removability is known. |
|  |  | All other masks are reserved and invalid for use |

### PARAMETERS

The table below lists the valid forms of the CID and the corresponding payload.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | NA | Empty | NA |
| Response | NA | MBIM\_SUBSCRIBER\_READY\_INFO\_EX3 | MBIM\_SUBSCRIBER\_READY\_INFO\_EX3 |

### RESPONSE

The following structure shall be used in the *InformationBuffer* of a query response. Compared to its counterpart in MBIMEx 2.0/1.0, the only difference is an added new field, **Flags**, right after the field ReadyState.

Table 3.4‑2: MBIM\_SUBSCRIBER\_READY\_INFO\_EX3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | ReadyState | MBIM\_SUBSCRIBER\_READY\_STATE | The same as in MBIM 1.0[1] |
| 4 | 4 | Flags | UINT32 | Bit maps of flags in MBIM\_SUBSCRIBER\_READY\_STATUS\_FLAGS |
| 8 | 4 | SubscriberIdOffset | OFFSET | The same as in MBIM 1.0[1] |
| 12 | 4 | SubscriberIdSize | SIZE (0..30) | The same as in MBIM 1.0[1] |
| 16 | 4 | SimIccIdOffset | OFFSET | The same as in MBIM 1.0[1] |
| 20 | 4 | SimIccIdSize | SIZE (0..40) | The same as in MBIM 1.0[1] |
| 24 | 4 | ReadyInfo | MBIM\_UNIQUE\_ID\_FLAGS | The same as in MBIM 1.0[1] |
| 28 | 4 | ElementCount (EC) | UINT32 | The same as in MBIM 1.0[1] |
| 32 | 8\*EC | TelephoneNumbersRefList | OL\_PAIR\_LIST | The same as in MBIM 1.0[1] |
| 32 + 8\*EC | 4 | DataBuffer | DATABUFFER | The same as in MBIM 1.0[1] |

When the ReadyState is MBIMSubscriberReadyStateInitialized, the modem shall set Flags field properly, based on whether the current SIM is eSIM profile or not and whether the current SIM is removable or embedded.

### NOTIFICATION

The data structure for notifications use the same data structure as for query response. The new Flags field, shall be set the same way as for query responses.

## MBIM\_CID\_MS\_MODEM\_CONFIG

### DESCRIPTION

This CID is used by an MBIM device to notify the host of the configuration progress and parameters of the device. It is also used by a host to query a MBIM device of the configurations parameters pre-configured in the device. These parameters are in general pre-configured in modem in factory or by OEM. They are stored in non-volatile memory, meant to be persistent, and do not change at run-time. They may consist of carrier/MO-specific parameters, generic and non-carrier-specific parameters and some OEM/device customization parameters.

When an MBIM device starts up or when SIM has changed, the device typically loads the configuration parameters that are appropriate to the current SIM. The host may use the indication of modem configuration status to provide the modem with registration parameters before the modem registers to the network.

When the MBIM device has detected a SIM and start loading of a particular configuration, it shall send an unsolicited MBIM\_CID\_MS\_MODEM\_CONFIG notification with ConfigStatus set to “Started”. All other fields will be ignored by the host. MBIM\_CID\_MS\_MODEM\_CONFIG notifications with ConfigStatus being “Completed” shall not be sent by the MBIM device until the SIM is initialized and SUPI/IMSI is read successfully.

Once the loading of a configuration has been completed, the MBIM device must notify the host by sending MBIM\_CID\_MS\_MODEM\_CONFIG with config status set to “Completed” . If config status is set to “Completed”, setting the config name is mandatory.

The maximum delay of between receiving the “Started” and “Completed” notifications at the host is 60seconds.

In certain situations, for example when device wakes up from the hibernate state, if the MBIM device has already initialize the SIM and completed the configuration before the MBIM interface is available then it is optional to queue the MBIM\_CID\_MS\_MODEM\_CONFIG notifications. The host will query the SIM status by sending an MBIM\_CID\_SUBSCRIBER\_READY\_STATUS query. If the MBIM device has not yet initialized the SIM, the host will turn the radio OFF to delay the initial registration.

If the ready state is set as “Initialized” and a new SIM is detected, the host will query modem configuration status by sending an MBIM\_CID\_MS\_MODEM\_CONFIG query. If the MBIM device has completed the configuration, it will return the response with Config Status set as “Completed”

Upon receiving a query request, if the device does not have its configuration ready, the device shall send a response indicating ConfigStatus as “Unknown” or “Started”.

Set requests and responses are not valid.

### PARAMETERS

The command code and associated service of this CID are defined in Table 3.1‑1 in Section 3.

The following table lists the valid forms of the CID and the corresponding payload for each legitimate form.

Table 3.5‑1: MBIM\_CID\_MS\_MODEM\_CONFIG\_INFO Payloads

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | NA | Empty | NA |
| Response | NA | MBIM\_MS\_MODEM\_CONFIG\_INFO | MBIM\_MS\_MODEM\_CONFIG\_INFO |

### DATA STRUCTURE

Table 3.5‑2: MBIM\_MS\_MODEM\_CONFIG\_STATUS

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMModemConfigStatusUnknown | 0 | Status is not known (e.g. SIM is not inserted) |
| MBIMModemConfigStatusStarted | 1 | SIM is ready and configuration is started |
| MBIMModemConfigStatusCompleted | 2 | SIM is ready and configuration is completed |

### QUERY

The information shall be set to null and the *InformationBufferLength* shall be zero.

### RESPONSE

Table 3.5‑3 shows the structure which shall be used in the *InformationBuffer* of a query response. The *InformationBufferLength* is variable. A Set response is not valid.

Table 3.5‑3: MBIM\_MS\_MODEM\_CONFIG\_ INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | ConfigStatus | MBIM\_MS\_MODEM\_CONFIG\_STATUS |  |
| 4 | Var | ConfigName | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | An informal name identifying the group of configuration items in effect for the current SIM. This name is informational for debugging purpose. It could be something like “AT&T” or “Rest of World”. The group of configuration items in effect may be an overlap of the carrier configuration for the SIM, some generic configuration items applied to all carriers, and some OEM/Engineering customization. It is not required that an unique name is generated whenever one item in the configuration is changed. |
| Var | Var |  | Unnamed IE’s | This payload may contain 0 or more unnamed and optional Information Elements (IE) encoded in MBIM\_TLV\_IE, at the end of the payload. Refer to section 3.2 for the format and processing rules. The presence or absence of TLV-typed IE of type MBIM\_TLV\_TYPE\_PRECFG\_DFLT\_CFG\_NSSAI has some specific meaning and restriction. See below for details. |

A TLV-typed IE of type MBIM\_TLV\_TYPE\_PRECFG\_DFLT\_CFG\_NSSAI may or may not be present as one of the unnamed IE’s. If such an IE is indeed present in a success notification or response with the config status being “Completed”, it contains the pre-configured default NSSAI in modem. If such an IE is absent in a success notification or response with the config status being “Completed”, the modem does not have pre-configured default NSSAI’s.

The maximum allowed response time of the device to a query request for this CID is 8 seconds. The device shall send a response within the maximum allowed response time. Late responses are invalid and will be discarded by the host.

### NOTIFICATION

The structure MBIM\_MS\_MODEM\_CONFIG\_ INFO shall be used in the *InformationBuffer* of notifications. The *InformationBufferLength* is variable.

### STATUS CODES

This CID only uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]). The following status codes are valid:

* MBIM\_STATUS\_SIM\_NOT\_INSERTED if SIM is not inserted.
* MBIM\_STATUS\_PIN\_REQUIRED if the SIM is locked.
* MBIM\_STATUS\_BAD\_SIM for other reasons where SIM is inserted but SUPI cannot be obtained.

## MBIM\_CID\_MS\_REGISTRATION\_PARAMS

### DESCRIPTION

This CID is used by the host to set 5G-specific registration parameters to an MBIM device. Before turning on the device radio, the host typically sends a set request of this CID to configure the device with desired registration parameters. These registration parameters include information such as default PDU session hint.

The host may send this SET command at any time. Upon receiving the command, the device shall compare the parameters to parameters used previously for 5G registration. If there are differences, the MBIM device shall use newly received parameters for next 5G registration. The set command also has a parameter to force immediate 5G re-registration.

The host may use this CID to query the registration parameters currently used by an MBIM device for 5G registration. Unsolicited notifications are not valid.

### PARAMETERS

The command code and belonging service of this CID are defined in Table 3.1‑1 in Section 3.

Table 3.6‑1 lists the valid forms of the CID and the corresponding payloads.

Table 3.6‑1: MBIM\_CID\_MS\_REGISTRATION\_PARAMS Payloads

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM\_MS\_REGISTRATION\_PARAMS\_INFO | Empty | NA |
| Response | MBIM\_MS\_REGISTRATION\_PARAMS\_INFO | MBIM\_MS\_REGISTRATION\_PARAMS\_INFO | NA |

### DATA STRUCTURES

Table 3.6‑2: MBIM\_MS\_MICO\_MODE

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMMicoModeDisabled | 0 | MICO mode is disabled on device |
| MBIMMicoModeEnabled | 1 | MICO mode is enabled on device |
| MBIMMicoModeUnsupported | 2 | Used in set or query response only, indicating that the device does not support MICO mode |
| MBIMMicoModeDefault | 3 | Used in set request only, indicating that the default MICO mode in the device (including no MICO mode support) be used by the device in 5G registration. |

Table 3.6‑3: MBIM\_MS\_DEFAULT\_PDU\_HINT

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMDefaultPDUActivationUnlikely | 0 | default PDU session activation is unlikely to happen soon. |
| MBIMDefaultPDUActivationLikely | 1 | default PDU session activation is likely to happen soon. |

Table 3.6‑4: MBIM\_MS\_LADN\_IND

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMLADNInfoNotNeeded | 0 | LADN information not needed |
| MBIMLADNInfoRequested | 1 | LADN information is requested |

Table 3.6‑5: MBIM\_MS\_REGISTRATION\_PARAMS\_ INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | MicoMode | MBIM\_MS\_MICO\_MODE | A value from Table 3.6‑2 representing Mobile Initiated Connection mode requested by the host.  In this MBIM Ex version, the host shall only set this field to MBIMMicoModeDisabled or MBIMMicoModeDefault. Other values are not to be used in set requests. |
| 4 | 4 | DRXParams | MBIM\_MS\_DRX\_PARAMS | A value from Table 4.1‑6 representing the configuration of DRX settings requested by the host.  In this MBIM Ex version, the host shall only set this field to MBIMDRXNotSpecified. Other values are not to be used in in set requests. |
| 8 | 4 | LADNInfo | MBIM\_MS\_LADN\_IND | A value from Table 3.6‑4 representing the LADN information requested by the host.  In MBIM Ex 3.0, the host shall only set this field to MBIMLADNInfoNotNeeded. Other values are not to be used in set requests. |
| 12 | 4 | DefaultPDUHint | MBIM\_MS\_DEFAULT\_PDU\_HINT | A value from Table 3.6‑3 representing a hint that there is possibility that the host will establish the default PDU session immediately after the registration is completed. However, since it depends on several policies, it may not happen. |
| 16 | 4 | ReRegisterIfNeeded | UINT32 | This is a forced re-registration indicator.  If the value is 1 and the device is currently registered or in the middle of registration, modem shall evaluate whether any of the registration parameters contained in this set request is different than those used in the previous registration request toward network. If yes, modem shall de-register from network and re-register using the current registration parameters. All normal procedures related to de-registration and registration, including sending appropriate events at appropriate times, shall be performed.  The value 0 indicates that forced re-registration is not requested.  All other values are reserved at this time.  This is a one-time indicator. Modem shall use this only once upon receiving this set request. |
| 20 | Var | var | Unnamed IE’s | This payload may contain 0 or more unnamed and optional Information Elements (IE) encoded in MBIM\_TLV\_IE, at the end of the payload. Refer to section 3.2 for the format and processing rules.  The presence or absence of TLV-typed IE of type MBIM\_TLV\_TYPE\_PRECFG\_DFLT\_CFG\_NSSAI has some specific meaning and restriction. See below and also in the sections of set/query request/response for details. |

A TLV-typed IE of type MBIM\_TLV\_TYPE\_PRECFG\_DFLT\_CFG\_NSSAI may or may not be present as one of the unnamed IE’s.

* If such an IE is present in a set request, the host instructs the device to use the contained NSSAI for 5G registration. In this MBIMEx version, it must exactly one S-NSSAI with SST being eMBB and no other elements if it is present.
* If such an IE is absent in a set request, the host don’t have preferred NSSAI for 5G registration and the device may choose any preferred NSSAI for 5G registration.
* If such an IE is present in a set or query response, it contains the device’s preconfigured default configured NSSAI that the device will use for 5G registration.
* If such an IE is absent in a set or query response, the device indicates that it does not have any preconfigured default configured NSSAI that the device will use for 5G registration.

### SET

The *InformationBuffer* shall contain the structure MBIM\_MS\_REGISTRATION\_PARAMS\_INFO defined in Table 3.6‑5**Error! Reference source not found.**. The parameters set in this structure, if accepted by the device, shall be used by the device during 5G registration requests.

If the device does no support MICO mode at all, the device may return MBIMMicoModeUnsupported in corresponding set response in an otherwise successful response. If the device does no support the specific MICO mode requested in a set request, the device may return the MICO mode that the device does support and will use for 5G registration in corresponding set response in an otherwise successful response.

If the device does no support setting DRX cycle at MBIM interface at all, the device may return MBIMDRXUnsupported in corresponding set response in an otherwise successful response. If the device does no support the specific DRX cycle requested in a set request, the device may return the DRX cycle that the device does support and will use for 5G registration in corresponding set response in an otherwise successful response. MBIMDRXNotSpecified can be used in set response to indicate that DRX cycle is unknown.

The parameter DefaultPDUHint contains a hint for whether default PDU session is imminent after 5G registration. The device may or may not use the same in 5G registration. The device should return the most likely value it will use in next 5G registration in the corresponding set response in an otherwise successful response.

The device, if able, must accept other parameters for 5G registration specified in a set request. Otherwise, the device shall fail the set request with an appropriate cause code.

The registration parameters last set by the host and accepted by the device remain in effect for 5G registration until the current SIM is removed or the device is rebooted. Upon SIM swap or device reboot, the host will set 5G registration parameters for the then current SIM for 5G registration. In case that the host does not, the device may use 5G registration parameters from the persistent modem configurations (if there) or whatever value as it sees fit.

MBIM devices shall maintain the persistent modem configuration and the host-settable 5G registration parameters independently, such that the set requests of this CID do not permanently change the persistent modem configuration.

### QUERY

The information shall be null and the *InformationBufferLength* shall be zero.

### RESPONSE

The structure MBIM\_MS\_REGISTRATION\_PARAMS\_INFO defined in Table 3.6‑5shall be used in the *InformationBuffer* of a successful set or query response. For failure set or query response, the information shall be null and the *InformationBufferLength* shall be zero.

In a successful set response, the structure shall contain the parameters set by the host and accepted by the device, as discussed in 3.6.4.

In a successful query response, the structure shall contain the parameters as below:

* If the parameters have been set by the host and accepted by the device since the device was rebooted or a different SIM was inserted, the structure shall contain the parameters set by the host and accepted by the device
* If the parameters have not been set by the host and accepted by the device since the device was rebooted or a different SIM was inserted, the structure shall contain the parameters that the device most likely will use for 5G registration.

### STATUS CODES

This CID only uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

## MBIM\_CID\_PACKET\_SERVICE

### DESCRIPTION

This CID is an extension of the preexisting MBIM\_CID\_PACKET\_SERVICE CID defined in Section 2.3. The extension adds Tracking Area Identity (TAI) and SubClass for MBIMDataClass5G which are defined in Section 3.2.

The MBIM device will report the TAI of currently registered Serving PLMN. The TAI field is valid only if the CurrentDataClass is set to MBIMDataClass5G and the CurrentDataSubClass indicates 5GC is in use. The MBIM device shall report a new TAI value each time the device moves to new cell having TAI different than previously reported TAI.

The SubClass field is used by the MBIM device to inform the host of the network configuration when registered to 5G. The CurrentDataSubClass field is valid only if the CurrentDataClass contains a data class that has valid subclasses and the possible values are defined in Table 3.3‑2**Error! Reference source not found.**.

### PARAMETERS

Table 3.7‑1: MBIM\_CID\_PACKET\_SERVICE Payloads

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM\_SET\_PACKET\_SERVICE | Empty | NA |
| Response | MBIM\_PACKET\_SERVICE\_INFO\_V3 | MBIM\_PACKET\_SERVICE\_INFO\_V3 | MBIM\_PACKET\_SERVICE\_INFO\_V3 |

### DATA STRUCTURES

### SET

The information is same as described in [1].

### QUERY

The information shall be null and the *InformationBufferLength* shall be zero.

### RESPONSE

The following structure shall be used in the *InformationBuffer*. Compared with the structure MBIM\_CID\_PACKET\_SERVICE\_INFO\_V2 defined in Section 2.5.6, the following structure has *SubClass* and *TrackingAreaIdentity* new fields.

Table 3.7‑2: MBIM\_PACKET\_SERVICE\_INFO\_V3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | NwError | UINT32 | A network-specific error code that comes from mobile operator network related to packet service. Possible error codes include, but not limited to, those  documented in the Cause values in the appendixes of the 3GPP TS 24.008 Specification. |
| 4 | 4 | PacketServiceState | MBIM\_PACKET\_SERVICE\_STATE | The same as in MBIM 1.0 |
| 8 | 4 | CurrentDataClass | MBIM\_DATA\_CLASS | The same as in MBIMEx 2.0 |
| 12 | 8 | UplinkSpeed | UINT64 | The same as in MBIM 1.0 |
| 20 | 8 | DownlinkSpeed | UINT64 | The same as in MBIM 1.0 |
| 28 | 4 | FrequencyRange | MBIM\_FREQUENCY\_RANGE | The same as in MBIM 2.0 |
| 32 | 4 | CurrentDataSubClass | MBIM\_DATA\_SUBCLASS | A bit from type MBIM\_DATA\_SUBCLASS defined in Table 3.3‑2 that represents the current data subclass related to the current data class in the field CurrentDataClass. If the CurrentDataCLass is MBIMDataClass5G, this field shall contain the appropriate bit for the 5G subclass. Otherwise, the value shall set to MBIMDataSubClassNone. |
| 36 | 8 | TrackingAreaIdentity | MBIM\_MS\_TAI | See Table 4.1‑3. All zeros in the MCC component of this field indicates unknown TAI. Note that 000 is not a valid and defined MCC. |
| 44 | Var | Unnamed IE’s | MBIM\_TLV\_IE | This payload may contain 0 or more unnamed and optional Information Elements (IE) encoded in MBIM\_TLV\_IE, at the end of the payload. Refer to section 3.2 for the format and processing rules. |

The maximum allowed response time by the modem to a query request of this CID is 8 seconds. The modem shall send a response within the maximum allowed response time. Late responses are invalid and will be discarded by the host.

### STATUS CODES

This CID uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

## MBIM\_CID\_MS\_NETWORK\_PARAMS

The definition of this CID is removed from this initial specification of MBIMEx version 3.0.

Note that the mechanism of adding and supporting optional CIDs in hosts and devices running a defined MBIMEx version exists in MBIM. This CID may be defined later and be supported in hosts and devices with MBIMEx version 3.0 at a later time.

## MBIM\_CID\_CONNECT

### DESCRIPTION

The description of this CID in original MBIM 1.0[1] applies, with the enhancements specified in this section.

For PDU sessions with context session type of either MBIMContextTypeIPv6 or MBIMContextTypeIPv4v6, the modem shall support SLAAC procedure for IPv6 stateless address autoconfiguration [9].

### PARAMETERS

The command code and the corresponding service for this CID are the same as in the original definition.

The valid forms of this CID are the same originally defined in [1]. The corresponding payloads are indicated in Table 3.9‑1.

Table 3.9‑1: MIBM\_CID\_CONNECT Payloads

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM\_SET\_CONNECT\_EX3 | MBIM\_CONNECT\_QUERY\_INFO\_EX3 | NA |
| Response | MBIM\_CONNECT\_INFO\_EX3 | MBIM\_CONNECT\_INFO\_EX3 | MBIM\_CONNECT\_INFO\_ EX3 |

MBIM\_set\_CONNECT\_EX3 is defined in 3.9.4. MBIM\_CONNECT\_INFO\_EX3 is defined in 3.9.5.

### DATA STRUCTURES

Table 3.9‑3: MBIM\_MS\_ACCESS\_MEDIA\_TYPE

|  |  |  |
| --- | --- | --- |
| Type names | Value | Description |
| MBIMAccessMediaTypeNone | 0 | No access media type preference is given or actual media type is unknown. |
| MBIMAccessMediaType3GPP | 1 | 3GPP access media. In requests, this indicates 3GPP access media only. |
| MBIMAccessMediaType3GPPPreferred | 2 | 3GPP access media is preferred |

### SET

The following structure shall be used in the *InformationBuffer* of a set request. The *InformationBufferLength* is variable.

Table 3.9‑4: MBIM\_SET\_CONNECT\_EX3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | SessionId | UINT32 | The same as in MBIM 1.0 |
| 4 | 4 | ActivationCommand | MBIM\_ACTIVATION\_COMMAND | The same as in MBIM 1.0 |
| 8 | 4 | Compression | MBIM\_COMPRESSION | The same as in MBIM 1.0 |
| 12 | 4 | AuthProtocol | MBIM\_AUTH\_PROTOCOL | The same as in MBIM 1.0 |
| 16 | 4 | ContextSessionType | MBIM\_CONTEXT\_IP\_TYPE | Session type (5G-NGC) or IP type (pre 5G-NGC) to request in session or context establishment |
| 20 | 16 | ContextPurposeType | MBIM\_CONTEXT\_TYPES | Specifies the type of context being represented in this connection, when the connection’s type is known, singular and significant. Host may specify MBIMContextTypeNone for any reason. This field is for informational purpose. The device shall not reject or fail a request solely based on the value in this field. |
| 36 | 4 | MediaPreference | MBIM\_MS\_ACCESS\_MEDIA\_TYPE | Preference of media or access type, such as Cellular (3GPP), WLAN (non-3GPP), or others, for the PDU session establishment request or PDP context activation request. The device may ignore it if not supported. |
| 40 | var | AccessString | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | For 5G NGC, this field contains the Data Network Name (DNN) in WCHAR string. For pre 5G NGC, it contains an APN.  The string may be an empty string, which indicates no DNN or APN is provided in the request  The size of the string must be less than or equal to 100 characters.  Other aspects remain the same as the AccessString in MBIM 1.0 |
| var | var | UserName | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | This field contains a UserName in wide-char string.  The size of the string must be less than or equal to 255 characters. |
| var | var | Password | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | This field contains a Password in wide-char string.  The size of the string must be less than or equal to 255 characters. |
| var | var | Unnamed IE’s | MBIM\_TLV\_IE | This payload may contain 0 or more unnamed and optional Information Elements (IE) encoded in MBIM\_TLV\_IE, at the end of the payload. Refer to 3.2 for the format and processing rules. |

When activating a PDP context or establishing a PDU session based on a set request of this CID, the modem shall ensure that the activated/established session has a context/session compatible with the type requested. Table 3.9‑5 shows the compatibility between activated IP type vs. requested IP type. The first column lists the possible requested context/session types in set requests for activation. Each of the remaining columns represents a context/session type in a successful response.

*Table 3.9‑5: Compatibility of requested session types vs activated session type*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Session types Requested | Default Type | IPv4 | IPv6 | IPv4v6 /  IPv4AndIPv6 | Ethernet | Unstructured |
| Default | Yes | Yes | Yes | Yes | No | No |
| IPv4 | No | Yes | No | No | No | No |
| IPv6 | No | No | Yes | No | No | No |
| IPv4v6 /  IPv4AndIPv6 | No | Yes | Yes | Yes | No | No |
| Ethernet | No | No | No | No | Yes | No |
| Unstructured | No | No | No | No | No | Yes |

If the activated/selected session type in an otherwise successful response to an activation request is incompatible to the requested session type, the host shall treat it as an invalid response and, at the host’s discretion, may take action to clean up the context/session.

### QUERY

The following structure defined in Table 3.9‑6, MBIM\_CONNECT\_QUERY\_INFO\_EX3, shall be used in the *InformationBuffer* of a query request.

*Table 3.9‑6: MBIM\_CONNECT\_QUERY\_INFO\_EX3*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | SessionId | UINT32 | Host specifies this member to uniquely identify the session for the context/session to query for |

### RESPONSE

The data structure defined in Table 3.9‑7, MBIM\_CONNECT\_INFO\_EX3, shall be used in the *InformationBuffer* of a set or query response.

*Table 3.9‑7: MBIM\_CONNECT\_INFO\_EX3*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | SessionId | UINT32 | The same as in MBIM 1.0 |
| 4 | 4 | ActivationState | MBIM\_ACTIVATION\_STATE | The same as in MBIM 1.0 |
| 8 | 4 | VoiceCallState | MBIM\_VOICE\_CALL\_STATE | The same as in MBIM 1.0 |
| 12 | 4 | ContextSessionType | MBIM\_CONTEXT\_IP\_TYPE | The session type (5G-NGC) or IP type (pre 5G-NGC) of the activated context or established session.  Note: when this data structure is used for payload of a deactivation response or of a notification for unsolicited deactivation, this field is optional and may contain  MBIMContextIPTypeDefault, in the event that the MBIM device no longer knowns the session (or IP) type of the corresponding PDU session (or PDP context). |
| 16 | 16 | ContextPurposeType | MBIM\_CONTEXT\_TYPES | Specifies the type of context being represented in this connection, when the connection’s type is known, singular and significant. Device may specify MBIMContextTypeNone for any reason. This field is for informational purpose. |
| 32 | 4 | NwError | UINT32 | A network-specific error code that comes from mobile operator network related to connection. Possible error codes include, but not limited to, those  documented in the Cause values in the appendixes of the 3GPP TS 24.008 Specification. |
| 36 | 4 | AccessMedia | MBIM\_MS\_ACCESS\_MEDIA\_TYPE | The selected media or access type for the established PDU session for 5G-NGC or the activated PDP context for pre 5G NGC.  The device may set it to NONE if the information is not available. |
| var | var | AccessString | MBIM\_TLV\_IE with type of MBIM\_TLV\_TYPE\_WCHAR\_STR | The DNN of the established PDU session for 5G-NGC or APN for the activated PDP context for pre 5G-NGC.  Note: when this data structure is used for payload of a deactivation response or a notification for unsolicited deactivation, this field is optional and may contain an empty string, in the event that the MBIM device no longer knowns the DNN (or APN) of the corresponding PDU session (or PDP context). |
| var | var | Unnamed IE’s | MBIM\_TLV\_IE | This payload may contain 0 or more unnamed and optional Information Elements (IE) encoded in MBIM\_TLV\_IE, at the end of the payload. Refer to 3.2 for the format and processing rules. |

### NOTIFICATION

The payload shall contain the same structure as that of set or query response.

### STATUS CODES

The device shall return status code MBIM\_CONTEXT\_SESSION\_TYPE\_NOT\_SUPPORTED if it does not support the requested ContextSessionType in this command.

The device shall return MBIM\_ACCESS\_MEDIA\_TYPE\_NOT\_SUPPORTED if it does not support the requested MediaPreference in this command.

For other status codes and their usage, see Microsoft Extensions to MBIM 1.0 Spec[1].

## MBIM\_CID\_IP\_PACKET\_FILTERS

### DESCRIPTION

The description of this CID in original MBIM 1.0 [1] applies. This extension spec adds a new member called “FilterId” to the data structure *MBIM\_SINGLE\_PACKET\_FILTER* of set command and response.

Both Query and Set contain an MBIM\_PACKET\_FILTERS in its InformationBuffer. For Query, however, the only relevant field is the SessionId. The SessionId in a Query indicates which IP data stream’s filters are to be returned by the device. MBIM\_PACKET\_FILTERS is returned from both Query and Set complete messages in the InformationBuffer.

### PARAMETERS

*Table 3.10‑1: MBIM\_CID\_PACKET\_FILTERS Payloads*

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM\_PACKET\_FILTERS | MBIM\_PACKET\_FILTERS | NA |
| Response | MBIM\_PACKET\_FILTERS | MBIM\_PACKET\_FILTERS | NA |

### DATA STRUCTURES

*Table 3.10‑2: MBIM\_SINGLE\_PACKET\_FILTER\_V2*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | FilterSize | SIZE (0..bMaxFilterSize) | The same as in MBIM 1.0 |
| 4 | 4 | PacketFilterOffset | OFFSET | The same as in MBIM 1.0 |
| 8 | 4 | PacketMaskOffset | OFFSET | The same as in MBIM 1.0 |
| 12 | 4 | FilterId | UINT32 | A FilterId contains a host-provided value that identifies the WOL pattern which can be used to identify wake reason by OS when incoming matched packet wake up host |
| 16 |  | DataBuffer | DATABUFFER | The same as in MBIM 1.0 |

### SET

The same MBIM\_PACKET\_FILTERS data structure as original MBIM 1.0[1] but with MBIM\_SINGLE\_PACKET\_FILTER\_V2 members replacing MBIM\_SINGLE\_PACKET\_FILTER. Inside MBIM\_PACKET\_FILTERS, each MBIM\_SINGLE\_PACKET\_FILTER is extended to have FilterId which will be used to identify wake reason by OS when device report wake reason. A host may set multiple IP packet filters to modem, each of which is identified by a unique FilterID. When the modem wakes from low power mode due to an incoming IP packet matching one of the IP packet filters, the modem shall remember the FilterID of the IP packet filter and shall return it upon wake reason query from the host via MBIM\_CID\_WAKE\_REASON.

### QUERY

The same MBIM\_PACKET\_FILTERS data structure as original MBIM 1.0[1].

### RESPONSE

See SET and Table 3.10-1: MBIM\_CID\_PACKET\_FILTERS Payloads. As in SET, the occurrences of MBIM\_SINGLE\_PACKET\_FILTER within the MBIM 1.0[1] definition are replaced by MBIM\_SINGLE\_PACKET\_FILTER\_V2.

### STATUS CODES

The same status codes as original MBIM 1.0[1].

## MBIM\_CID\_MS\_WAKE\_REASON

### DESCRIPTION

This is an optional new CID. If a device advertises it supports this function, host will query the device its wake reason when power up because of device wake. If the device supports this function, but fails the query from host, it means the power up is not caused by a wake from the device.

The device needs to save the wake reason until it receives MBIM\_CID\_MS\_WAKE\_REASON query from OS, if this optional CID is supported by modem.

### PARAMETERS

*Table 3.111‑1: MBIM\_CID\_WAKE\_REASON Payloads*

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | NA | Empty | NA |
| Response | NA | MBIM\_WAKE\_REASON | NA |

### DATA STRUCTURES

*Table 3.111‑1: MBIM\_WAKE\_TYPE*

|  |  |  |
| --- | --- | --- |
| Type names | Value | Description |
| MBIMWakeTypeCIDResponse | 0 | A CID response wakes device |
| MBIMWakeTypeCIDIndication | 1 | A CID indication wakes device |
| MBIMWakeTypePacket | 2 | An incoming packet wakes device |

### QUERY

The InformationBuffer shall be null and InformationBufferLength shall be zero

### RESPONSE

The following structure shall be used in the InformationBuffer. According to the WakeType, the first TLV is either WakeMessage or WakePacket. The response cannot have both.

*Table 3.111‑3: MBIM\_WAKE\_REASON*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | WakeType | UINT32 | See Table 3.111‑2: MBIM\_WAKE\_TYPE |
| 4 | 4 | SessionId | UINT32 | Specifies the session that wakes the device |
| 8 | var | DataBuffer | Byte array | If WakeType contains either MBIMWakeTypeCIDResponse or MBIMWakeTypeCIDIndication , DataBuffer must contain a TLV for wake message, defined in Table 3.1110: MBIM\_MS\_WAKE\_COMMAND. If WakeType contains MBIMWakeTypePacket, DataBuffer must contain a TLV for wake packet defined in Table 3.1111: MBIM\_MS\_WAKE\_PACKET. |

### STATUS CODES

This CID uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]). A failure code means the device is not powered up by a wake.

## MBIM\_CID\_BASE\_STATIONS\_INFO

### DESCRIPTION

The description of this CID in the original MBIM Extension specification [12].applies, with extension to support 5G NR cell measurements for Location Service below.

5G NR serving cells include one primary cell and optionally one or more secondary cells. In case there are multiple NR serving cells, the registered cell with the device must be the first entry in that list.

The neighbor cells are applicable for either NR cells or EUTRA cells, as described in [[11]](#_References).

### PARAMETERS

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | NA | MBIM\_BASE\_STATIONS\_INFO\_REQ | NA |
| Response | NA | MBIM\_BASE\_STATIONS\_INFO | NA |

### QUERY

The following structure shall be used in the InformationBuffer, this is used to configure aspects of the cell information such as the maximum number of serving cells measurements to send in response.

*Table 3.12‑1: MBIM\_BASE\_STATIONS\_INFO\_REQ*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | MaxGSMCount | SIZE | Same as in previous MBIM version. |
| 4 | 4 | MaxUMTSCount | SIZE | Same as in previous MBIM version. |
| 8 | 4 | MaxTDSCDMACount | SIZE | Same as in previous MBIM version. |
| 12 | 4 | MaxLTECount | SIZE | Same as in previous MBIM version. |
| 16 | 4 | MaxCDMACount | SIZE | Same as in previous MBIM version. |
| 20 | 4 | MaxNRCount | SIZE | Maximum number of entries of both 5G NR serving cells and neighbor cells returned in measurement results. This is applicable for MBIMDataClass5G with any data subclass that involves NR.  The serving cells measured results list in **MBIM\_NR\_SERVING\_CELLS\_INFO**; and the neighbor cells measured results list in **MBIM\_NR\_NEIGHBOR\_CELLS\_INFO**.  Default capacity is 40, with maximum 32 serving cells and 8 best NR or EUTRA neighbor cells. |

### RESPONSE

The following structure shall be used in the InformationBuffer of the MBIM\_COMMAND\_DONE.

*Table 3.12‑2: MBIM\_BASE\_STATIONS\_INFO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | SystemType | MBIM\_DATA\_CLASS | Indicates the system type or types for which serving cell information is valid, a bitmask of one or more system types as defined in the MBIM\_DATA\_CLASS |
| 4 | 4 | SystemSubType | MBIM\_DATA\_SUBCLASS | Indicates the 5G data subclass for which 5G serving cell information is valid, a bitmask of one or more system types as defined in the MBIM\_DATA\_SUBCLASS. Only valid when the SystemType field above indicates that 5G serving cell information is valid. Should be MBIMDataSubClassNone otherwise. |
| 8 | 4 | GSMServingCellOffset | OFFSET | Same as in previous MBIM version. |
| 12 | 4 | GSMServingCellSize | SIZE(0..44) | Same as in previous MBIM version. |
| 16 | 4 | UMTSServingCellOffset | OFFSET | Same as in previous MBIM version. |
| 20 | 4 | UMTSServingCellSize | SIZE(0..60) | Same as in previous MBIM version. |
| 24 | 4 | TDSCDMAServingCellOffset | OFFSET | Same as in previous MBIM version. |
| 28 | 4 | TDSCDMAServingCellSize | SIZE(0..48) | Same as in previous MBIM version. |
| 32 | 4 | LTEServingCellOffset | OFFSET | Same as in previous MBIM version. |
| 36 | 4 | LTEServingCellSize | SIZE(0..48) | Same as in previous MBIM version. |
| 40 | 4 | GSMNmrOffset | OFFSET | Same as in previous MBIM version. |
| 44 | 4 | GSMNmrSize | SIZE | Same as in previous MBIM version. |
| 48 | 4 | UMTSMrlOffset | OFFSET | Same as in previous MBIM version. |
| 52 | 4 | UMTSMrlSize | SIZE | Same as in previous MBIM version. |
| 56 | 4 | TDSCDMAMrlOffset | OFFSET | Same as in previous MBIM version. |
| 60 | 4 | TDSCDMAMrlSize | SIZE | Same as in previous MBIM version. |
| 64 | 4 | LTEMrlOffset | OFFSET | Same as in previous MBIM version. |
| 68 | 4 | LTEMrlSize | SIZE | Same as in previous MBIM version. |
| 72 | 4 | CDMAMrlOffset | OFFSET | Same as in previous MBIM version. |
| 76 | 4 | CDMAMrlSize | SIZE | Same as in previous MBIM version. |
| 80 | 4 | NRServingCellsOffset | OFFSET | Offset in bytes, calculated from beginning of this structure, to the buffer containing NR Measure results list.  This member can be zero when no NR serving cells available for device without 5G capability. |
| 84 | 4 | NRServingCellsSize | SIZE | Total size in bytes of the buffer containing 5G NR measured results in format of MBIM\_NR\_SERVING\_CELLS |
| 88 | 4 | NRNeighborCellsOffset | OFFSET | Offset in bytes, calculated from beginning of this structure, to the buffer containing NR neighbor cells measurement results.  This member can be NULL when no NR neighbor cells available. |
| 92 | 4 | NRNeighborCellsSize | SIZE | Total size in bytes of the buffer containing NR measured results in format of MBIM\_NR\_NEIGHBOR\_CELLS |
| 96 |  | DataBuffer | DATABUFFER | GSMServingCell  UMTSServingCell  TDSCDMAServingCell  LTEServingCell  GSMNmr  UMTSMrl  TDSCDMAMrl  LTEMrl  CDMAMrl  NRServingCells  NRNeighborCells |

*Table 3.12-3: MBIM\_NR\_SERVING\_CELLS*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | ElementCount(EC) | UINT32 | Count of NR serving cell entries following this element, with maximum of 32 NR serving cells. |
| 4 |  | DataBuffer | DATABUFFER | Array of NR serving cell records, each specified as MBIM\_NR\_SERVING\_CELLS\_INFO |

*Table 2‑4: MBIM\_NR\_SERVING\_CELLS\_INFO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | ProviderIdOffset | OFFSET | Offset in bytes, calculated from the beginning of this structure, to a numeric (0-9) string ProviderId that represents the network provider identity.  This string is a concatenation of a three-digit Mobile Country Code (MCC) and a two or three-digit Mobile Network Code (MNC). This member can be NULL when no ProviderId information is returned. |
| 4 | 4 | ProviderIdSize | SIZE(0..12) | Size used for the ProviderId |
| 8 | 8 | NCI | UINT64 | NR Cell Identity is a 36-bit identity having value range (0..68719476736). NR Cell Global Identity (NCGI) can be constructed from ProviderId and NCI. Use 0xffffffffffffffff when the information is not available. |
| 16 | 4 | PhysicalCellID | UINT32 | NR Physical CellID (0..1007). Use 0xffffffff when the information is not available. |
| 20 | 4 | NRARFCN | UINT32 | Absolute Radio Frequency Channel Number of serving cell (0..3279165). Use 0xffffffff when the information is not available. |
| 24 | 4 | TAC | UINT32 | Tracking Area Code is a 24-bit value range (0..16777215), which is used to identify tracking area within scope of a PLMN. The absence of TAC field indicates Cell only supports EN-DC functionality. Use 0xffffffff when the information is not available. |
| 28 | 4 | RSRP | UINT32 | Average Reference Signal Received Power. Range (0 .. 127) as defined in [[11]](#_References), and mapped to (156.. 31) dBm, as defined in [[10]](#_References). Use 0xffffffff when the information is not available. |
| 32 | 4 | RSRQ | UINT32 | Average Reference Signal Received Quality. Range (0 .. 127) as defined in [[11]](#_References), and mapped to (43 .. 20) dB, defined in [[10]](#_References). Use 0xffffffff when the information is not available. |
| 36 | 4 | SINR | UINT32 | Average Reference Signal to Noise and Interference Ratio. Range (0 .. 127) as defined in [[11]](#_References), and mapped to (-23 .. 40 ) dB, defined in [[10]](#_References). Use 0xffffffff when the information is not available. |
| 40 | 8 | TimingAdvance | UINT64 | Timing advance value in microseconds, where formula is defined in section 7.2 in [10]. Use 0xffffffff when the information is not available.. |
| 48 |  | DataBuffer | DATABUFFER | ProviderId |

*Table 3.12-5: MBIM\_NR\_NEIGHBOR\_CELLS*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | ElementCount(EC) | UINT32 | Count of neighbor cell entries following this element, with maximum of 8 best NR or EUTRA neighbor cells. |
| 4 |  | DataBuffer | DATABUFFER | Array of NR serving cell records, each specified as MBIM\_NR\_NEIGHBOR\_CELLS\_INFO |

*Table 3.12‑6: MBIM\_NR\_NEIGHBOR\_CELLS\_INFO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | SystemSubType | MBIM\_DATA\_SUBCLASS | Indicates the system type for which a neighbor cell information is valid. This is applicable for indicating whether neighbor cell is either NR or EUTRA cell corresponding to the types defined in MBIM\_DATA\_SUBCLASS. |
| 4 | 4 | ProviderIdOffset | OFFSET | Offset in bytes, calculated from the beginning of this structure, to a numeric (0-9) string ProviderId that represents the network provider identity.  This string is a concatenation of a three-digit Mobile Country Code (MCC) and a two or three-digit Mobile Network Code (MNC). This member can be NULL when no ProviderId information is returned. |
| 8 | 4 | ProviderIdSize | SIZE(0..12) | Size used for the ProviderId |
| 12 | 4 | CellIDOffset | OFFSET | Offset in bytes, calculated from beginning of structure. This member can be NULL when no CellID information is available. |
| 16 | 4 | CellIDSize | SIZE(0..7) | Size used for CellID, which is 28-bit string. |
| 20 | 4 | PhysicalCellID | UINT32 | Physical CellID (0..1007). Use 0xffffffff when the information is not available. |
| 24 | 4 | TAC | UINT32 | Tracking Area Code is a 24-bit value for EUTRA-5GC or 16-bit value for EUTRA-EPC, which is used to identify tracking area within scope of a PLMN. Use 0xffffffff when the information is not available. |
| 28 | 4 | RSRP | UINT32 | Average Reference Signal Received Power. Range (0 .. 127) as defined in [[11]](#_References), and mapped to (156.. 31) dBm, as defined in [[10]](#_References). Use 0xffffffff when the information is not available. |
| 32 | 4 | RSRQ | UINT32 | Average Reference Signal Received Quality. Range (0 .. 127) as defined in [[11]](#_References), and mapped to (43 .. 20) dB, defined in [[10]](#_References). Use 0xffffffff when the information is not available. |
| 36 | 4 | SINR | UINT32 | Average Reference Signal to Noise and Interference Ratio. Range (0 .. 127) as defined in [[11]](#_References), and mapped to (-23 .. 40 ) dB, defined in [[10]](#_References). Use 0xffffffff when the information is not available. |
| 40 |  | DataBuffer | DATABUFFER | ProviderId  CellID |

## MBIM\_CID\_MS\_LTE\_ATTACH\_STATUS

### DESCRIPTION

All aspects of this CID are inherited from MBIMEx 2.0 and remain the same, except for those explicitly modified in this section, 3.13.

In MBIMEx 3.0, this CID is modified to be able to convey the error cause code, if available, for LTE attach/detach related procedures.

### PARAMETERS

In MBIMEx 3.0, the payload of query responses and notifications shall use a new data structure, MBIM\_MS\_LTE\_ATTACH\_STATUS\_V2, defined in 3.13.3.

*Table 3.13‑1: MBIM\_CID\_PACKET\_FILTERS Payloads*

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | N/A | N/A | N/A |
| Response | N/A | MBIM\_MS\_LTE\_ATTACH\_STATUS\_V2 | MBIM\_MS\_LTE\_ATTACH\_STATUS\_V2 |

### DATA STRUTURES

In MBIMEx 3.0, the data structure below, MBIM\_MS\_LTE\_ATTACH\_STATUS\_V2, is defined for the payload for query responses and notifications. Comparing the corresponding data structure in MBIMEx 2.0 (MBIM\_MS\_LTE\_ATTACH\_STATUS), the only difference is a new field, NwError.

*Table 3.13‑2: MBIM\_MS\_LTE\_ATTACH\_STATUS\_V2*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | LteAttachState | MBIM\_MS\_LTE\_ATTACH\_STATE | The same as that in effect in MBIMEx 2.0 |
| 4 | 4 | NwError | UINT32 | An error cause code that is related to LTE attachment/detachment procedures. It could be a network-provided error cause code or an internal error cause code.  Network-provided error cause codes include, but not limited to, the EMM Cause values documented in 3GPP TS 24.301 and shall be in the range of [0x1 – 0x7FFFFFFF]  Internal error cause codes are specific to a device’s implementation for information purpose and shall be in the range of [0x80000000 – 0xFFFFFFFE].  Special code 0 indicates that there is no error. Special code 0xFFFFFFFF indicates that the error cause error is unknown to the device when sending a response or a notification. |
| 8 | 4 | IPType | MBIM\_CONTEXT\_IP\_TYPES | The same as that in effect in MBIMEx 2.0 |
| 12 | 4 | AccessStringOffset | OFFSET | The same as that in effect in MBIMEx 2.0 |
| 16 | 4 | AccessStringSize | SIZE(0..200) | The same as that in effect in MBIMEx 2.0 |
| 20 | 4 | UserNameOffset | OFFSET | The same as that in effect in MBIMEx 2.0 |
| 24 | 4 | UserNameSize | SIZE(0..510) | The same as that in effect in MBIMEx 2.0 |
| 28 | 4 | PasswordOffset | OFFSET | The same as that in effect in MBIMEx 2.0 |
| 32 | 4 | PasswordSize | SIZE(0..510) | The same as that in effect in MBIMEx 2.0 |
| 36 | 4 | Compression | MBIM\_COMPRESSION | The same as that in effect in MBIMEx 2.0 |
| 40 | 4 | AuthProtocol | MBIM\_AUTH\_PROTOCOL | The same as that in effect in MBIMEx 2.0 |
| 44 |  | DataBuffer | DATABUFFER | The same as that in effect in MBIMEx 2.0 |

A device shall report the error cause code related to LTE attach/detach procedures at the earliest possible instance, either in a query response being sent or by generating a unsolicited notification. A device shall also retain the latest error cause code related to LTE attach/detach procedures and return it in response to next query request from the host.

# MBIM Interface Extensions for 5G NGC – Phase 2

This section defines the MBIM extensions for second phase of features for 5G system with the next generation core network (NGC, aka 5GC), including deployment options 2, 4, 7 and 5. These extensions build on top of the MBIM extensions for 5G NGC – Phase 1 defined in Section 3.

## Versioning Scheme

A new MBIM Extensions Release number 4.0 is introduced for the changes specified in Section 4. Those changes include breaking changes from the existing MBIM Extensions Release numbers 1.0, 2.0, and 3.0.

As in Section 2.1, the term “MBIMEx version” refers to the MBIM Extensions Release number of this document.

MBIMEx version 4.0 is under existing MBIM Release number 1.0, like MBIMEx version 1.0 [1], MBIMEx version 2.0, and MBIMEx version 3.0.

Unless explicitly documented otherwise, all procedures for versioning defined in Section 2.1 apply to both host and device. A host and a device shall follow those procedures to learn one another’s version capabilities and to subsequently agree to an MBIMEx version to govern the MBIM interface for the duration of the device being enumerated to the host.

A device capable of MBIMEx 4.0 with intention to work with older hosts should initially advertise MBIMEx version 1.0 in MBIM extended functional descriptor. If the host sends MBIM\_CID\_VERSION and the host has an MBIMEx version no less than 4.0, the device should, in the MBIM\_CID\_VERSION response, indicates MBIMEx 4.0. At that point, the host and the device agree to use MBIMEx 4.0.

Unless explicitly mentioned and modified, all unmentioned payloads, CIDs and procedures in MBIMEx versions 3.0, 2.0 and 1.0 carry over and stay unchanged in MBIMEx version 4.0.

## MBIM\_CID\_MS\_AUTHENTICATE

### DESCRIPTION

This CID is used by an MBIM device and the host during EAP authentication. When the MBIM device receives the first EAP request from the network, it will set the Session ID field associated the current PDU session and set the sequence number to 1. The message type is MBIMAuthMsgCommand and EAP content shall be copied to EapPAcket field. The PCO field is optional.

The MBIM host will then start the authentication process and send EAP response on MBIM\_CID\_MS\_AUTHENTICATE Set command to the MBIM Device. It shall use the same SessionID it received from the MBIM device. The sequence number shall set same as correspondence EAP request message. The message type is MBIMAuthMsgResponse and EAP response shall be copied to EAPPacket field. The PCO field is optional.

If any further EAP Request/Responses are needed during the authentication, then MBIM device and shall increase the sequence number by 1 each time it transfers the EAP packet.

Finally, when the authentication server informs a successful authentication, the MBIM device shall use message type as MBIMAuthMsgTypeSuccess. The sequence number shall be increased by 1 from the last sent MBIMAuthMsgTypeCommand message.

Anytime, if the authentication server cannot authenticate, the MBIM device shall use message type as MBIMAuthMsgTypeFailure.

If the sequence numbers received from the MBIM modem are not sequential, the MBIM host ignore the out of sequence messages.

Query requests and responses are not valid.

### PARAMETERS

The command code and associated service of this CID are defined in Table 3.1‑1 in Section 3.

The following table lists the valid forms of the CID and the corresponding payload for each legitimate form.

Table 3.2‑1: MBIM\_MS\_AUTH\_INFO Payloads

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | MBIM \_MS\_AUTH\_INFO | NA | NA |
| Response | Empty | NA | MBIM\_MS\_AUTH\_INFO |

### DATA STRUCTURE

Table 3.2‑2: MBIM\_MS\_AUTH\_MSG\_TYPE

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMAuthMsgTypeUnknown | 0 | Message is unknown |
| MBIMAuthMsgTypeRequest | 1 | Message contains EAP request |
| MBIMAuthMsgTypeResponse | 2 | Message contains EAP response |
| MBIMAuthMsgTypeSuccess | 3 | Message contain successful authentication results |
| MBIMAuthMsgTypeFailure | 4 | Message contain failed authentication results |

### SET

The following structure shall be used in the *InformationBuffer* of a set request. The *InformationBufferLength* is variable.

Table 3.12-3: MBIM\_MS\_AUTH\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | SessionID | UINT32 | The host specifies this member to uniquely identify the session for PDU session |
| 4 | 4 | SeqNum | UINT32 | A unique session specific sequence number. Each EAP pair, request and response, shall be using same sequence number. Sequence number pair shall be sequential on a specific session. |
| 8 | 4 | MsgType | MBIM\_MS\_AUTH\_MSG\_TYPE | Message type specifies the type of EAP packet. See Table 3.12-2 for possible packet types. |
| 12 | Var | EapPacket | MBIM\_TLV\_TYPE\_EAP\_PACKET | An EAP packet defined in Table 3.2‑2. |
|  | Var | Pco | MBIM\_TLV\_TYPE\_PCO | A PCO container defined in Table 3.2-2. |

### RESPONSE

The InformationBuffer shall be null and InformationBufferLength shall be zero

### NOTIFICATION

See Table 3.12.4-1: MBIM\_MS\_AUTH\_INFO

### STATUS CODES

This CID only uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

## MBIM\_CID\_MS\_NETWORK\_PARAMS

### DESCRIPTION

This CID is used by the host to retrieve the network parameters from an MBIM device. The network may send configuration and policies to the UE at any time. For example, after registration or when moving to a different tracking area in 5G network, the 5GC may use the Configuration Update procedure to send configurations to the UE. Such configuration data may include allowed NSSAI and rejected NSSAI. For another example, 5GC may use the MANAGE UE POLICY procedure to send URSPs to UE any time after registration.

Whenever an MBIM device receives such configuration data and/or policy information, the device shall use unsolicited notifications of this CID to notify the host of these new configurations and policies that are reported using this CID.

The host may use a query request of this CID to retrieve the current configurations and policies. Upon receiving a query request, the device shall respond with the latest network parameters that are reported using this CID.

#### Network Parameter Groups

The network parameters are grouped into parameter groups. A group of network parameters typically can be independently set by network, and therefore can change independently. The device may send unsolicited notifications per a single group or multiple groups. Host may also query the parameters per a single group or multiple groups. Currently, two parameter groups are identified for 5G:

* Configuration Group. This group includes all parameters that are closely related to the parameters in the Configuration Update procedure
* Network-set UE Policies Group. This group includes URSP and ANDSP that can change in UE per the MANAGE UE POLICY procedure

##### Configurations Group

The parameters in this group include those that can be updated by network in 5G via the following procedures/messages:

* UE Configuration update procedure (See [5], Section 5.4.4). This can update a wide range of parameters such as TAI list, LADN information, Allowed NSSAI, Rejected NSSAI and Configured NSSAI.
* RegistrationAccept message in Registration procedure (See [5], Section 5.5.1). It can update such parameters as Allowed NSSAI, Rejected NSSAI and Configured NSSAI.
* UE Parameters Update via UDM Control Plane Procedure (See [4], Section 4.20.2, and [5], Section 9.11.3.53A). It can update such parameter as Default Configured NSSAI.

The Network Parameter Group for Configuration may include one or more information elements in Table 3.8‑1.

Table 3.8‑1: Network Configuration Information Elements

|  |  |  |  |
| --- | --- | --- | --- |
| Size | Information Element | Type | Description |
| 4 | MICO indication | MBIM\_MS\_MICO\_IND | Indicates the use of MICO mode or the re-negotiation of MICO mode |
| 4 | Negotiated DRX parameters | MBIM\_MS\_DRX\_PARAMS | Indicates the DRX cycle value to be used at paging |
| Variable | TAI list | MBIM\_TLV\_TYPE\_TAI | List of tracking areas |
| Variable | Allowed NSSAI | MBIM\_TLV\_TYPE\_ALLOWED\_NSSAI | Set of S-NSSAIs that can be used in the current registration area |
| Variable | Configured NSSAI | MBIM\_TLV\_TYPE\_CFG\_NSSAI | Set of provisioned S-NSSAIs for one or more PLMN |
| Variable | Rejected NSSAI | MBIM\_TLV\_TYPE\_REJ\_NSSAI | Set of S-NSSAIs not allowed in the current PLMN or registration area |
| Variable | Default Configured NSSAI | MBIM\_TLV\_TYPE\_DFLT\_CFG\_NSSAI | Default S-NSSAI(s) used for initial access |
| Variable | LADN information | MBIM\_TLV\_TYPE\_LADN | Specifies the LADN service area for each available LADN in the current registration area |

##### Network-set UE Policies Group

The parameters in this group are the URSP and ANDSP that can be updated by network in 5G via the following procedures/messages:

* Network-requested UE policy management procedure (See [5], Section D.2.1)

The home PLMN may provide UE policies to the UE by the procedure above. It is the modem’s responsibility to store the UE policies in a non-volatile memory together with the SUPI from the SIM. The UE policies shall remain intact during reboot of the modem and/or reboot of the system containing the host and modem. The UE policies can only be used if the SUPI from the SIM matches the SUPI stored in the device’s non-volatile memory with the URSP. If the SUPI from the SIM does not match the SUPI stored in the non-volatile memory, the modem shall invalidate the UE policies and cease to report the invalid values to the host.

When the SIM in the modem is removed, the modem shall invalidate the UE policies associated with the SUPI of the removed SIM and cease to report the invalid values to the host.

Upon receipt of the MANAGE UE POLICY COMMAND message during a network-requested UE policy management procedure, the modem processes all UE policy sections from the message. Each UE policy section is identified by a UE policy section identifier (UPSI). A UPSI is composed of a PLMN ID and a UE policy section code (UPSC). Refers to [5], Section D.1. For each UE policy section, the modem shall:

* store the received UE policy section if the modem has no stored UE policy section associated with the same UPSI as the UPSI associated with the instruction;
* replace the stored UE policy section with the received UE policy section if the modem has a stored UE policy section associated with the same UPSI as the UPSI associated with the received section; or
* delete the stored UE policy section if the modem has a stored UE policy section associated with the same UPSI as the UPSI associated with the instruction and the UE policy section contents are empty.

After processing the MANAGE UE POLICY COMMAND message, the modem shall send a notification of this CID with the current UE policies.

Upon receiving a query request for UE policies, the modem shall send a response with the current UE policies.

A deleted UE policy section or a section with empty UE policy section contents is not considered a part of the UE policies. If UE policy section with empty UE policy section contents is included in a notification or query response, it will be ignored by the host.

When sending UE policies in responses and notification, the UE policies shall be encoded within an MBIM\_TLV\_IE structure with the type of MBIM\_TLV\_TYPE\_UE\_POLICIES (see 3.2 for MBIM\_TLV\_IE). Its Data field shall contain “UE policy section management list contents” whose structure and encodings are as defined in [5] TS 24.501 “Non-Access Stratum (NAS) Protocol for 5G System (5GS); Stage 3”, Section D.6.2 “UE policy section management list”, and in [6] TS 24.526 “Policies for 5G System; Stage 3”, Section 5 “Encoding of UE Policies”. As defined there, “UE policy section management list contents” has the following overall structure:

Figure 4: UE policy section management list contents

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |  |
| UE policy section management sublist (PLMN 1) | | | | | | | | octet 1  octet a |
| UE policy section management sublist (PLMN 2) | | | | | | | | octet a+1  octet b |
| … | | | | | | | | octet b+1  …  octet c |
| UE policy section management sublist (PLMN N) | | | | | | | | octet c+1  octet z |

If the modem does not support including the “Route Selection Descriptors”, also defined in [5] TS 24.501 as part of the MBIM\_TLV\_TYPE\_UE\_POLICIES it must not set MBIMDeviceCapsUEPolicyRouteSelection in MBIM\_CID\_DEVICE\_CAPS\_V2. Further, the modem must account for the absence of the “Route Selection Descriptors” and set the length fields as appropriate, taking into account this data is absent, within in each of the structures which make up this payload.

### PARAMETERS

The command code and belonging service of this CID are defined in Table 3.1‑1 in Section 3.

The following table indicates the valid forms of the CID and the corresponding payload for each valid form.

Table 3.8‑2: MBIM\_CID\_MS\_NETWORK\_PARAMS Payloads

|  |  |  |  |
| --- | --- | --- | --- |
|  | Set | Query | Notification |
| Command | NA | MBIM\_MS\_NW\_PARAMS\_QUERY\_INFO | NA |
| Response | NA | MBIM\_MS\_NW\_PARAMS\_INFO | MBIM\_MS\_NW\_PARAMS\_INFO |

MBIM\_MS\_NW\_PARAMS\_QUERY\_INFO is defined in Table 3.8‑6. MBIM\_MS\_NW\_PARAMS\_INFO is defined in Table 3.8‑7.

Set requests and responses are not valid for this CID.

### DATA STRUCTURES

Table 3.83 defines the Rejected S-NSSAI data format for MBIM\_TLV\_TYPE\_REJ\_NSSAI.

The TLV Data section must include one or more Rejected S-NSSAIs. Each rejected S-NSSAI consists of one S-NSSAI and an associated cause value.

Table 3.8‑3: MBIM\_MS\_REJ\_SNSSAI\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 1 | SnssaiLength | UINT8 | Length is expressed per 3GPP TS 24. 501 section 9.11.2.8.  The length of the included S-NSSAI contents in octets.  Depending on the value of the length field the following S-NSSAI contents are included: 1 - SST 4 - SST and SD All other values are reserved |
| 1 | 1 | Cause | UINT8 | Cause Value 0 - S-NSSAI not available in the current PLMN 1 - S-NSSAI not available in the current registration area All other values are reserved |
| 2 | 1 | Sst | UINT8 | Slice/service type (SST) 1 - Enhanced Mobile Broadband 2 - Ultra-reliable low latency communications. 3 - Massive IoT All other values are reserved |
| 3 | 3 | Sd | UINT8 | Slice differentiator (SD) Only included when SnssaiLength equal to 4 |

Table 3.84 defines the LADN data format for the TLV type MBIM\_TLV\_TYPE\_LADN. The TLV Data section must include one or more LADN values.

Table 3.8‑4: MBIM\_MS\_LADN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | Variable | Dnn | MBIM\_MS\_DNN | DNN string |
| Variable | Variable | TaiList | MBIM\_MS\_TAI\_LIST\_INFO | Tracking area identity list from Table 3.2‑3 |

Table 3.8‑5 defines the Registration Area Allocation Indication (RAAI) values used to indicate the MICO mode negotiated with the network. If the device sends a MICO mode request and it is supported and accepted by the network, the network will include a MICO indication response when accepting the registration procedure.

Table 3.8‑5: MBIM\_MS\_MICO\_IND

|  |  |  |
| --- | --- | --- |
| Type | Value | Description |
| MBIMRaaiTypeRaNotAllocated | 0 | All PLMN registration area not allocated |
| MBIMRaaiTypeRaAllocated | 1 | All PLMN registration area allocated |
| MBIMRaaiTypeNotAvailable | 0xffffffff | RAAI not received from the network. Either the device did not send a MICO mode request, MICO mode is not supported by the network, or device is not registered. |

### QUERY

The data format in Table 3.8‑6 shall be used in the *InformationBuffer* of a query request. The data indicates which part or parts of the network parameters are to be queried. The *InformationBufferLength* is variable.

Table 3.8‑6: MBIM\_MS\_NW\_PARAMS\_QUERY\_INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 2 | AreConfigurationsNeeded | UNIT16 | Value 0 indicates that network parameters of the Configurations group are not needed; Other values indicates that they are needed. |
| 2 | 2 | AreUEPoliciesNeeded | UNIT16 | Value 0 indicates that the network parameters of the UE Policies group are not needed;  Other values indicate that they are needed. |
| 4 | variable | Unnamed IE’s | octets | This payload may contain 0 or more unnamed and optional Information Elements (IE) encoded in MBIM\_TLV\_IE. Refer to 3.2 for the format and processing rules. Optional unnamed IE’s in MBIM\_TLV\_IE format. |

If network parameters of the Configurations group are needed, as indicated by the field AreConfigurationsNeeded, and if the query is successful, the modem shall return the current network parameters of the Configurations group in the response.

If network parameters of the Network-set UE Policies group are needed, as indicated by the field AreUEPoliciesNeeded, and if the query is successful, the modem shall return the current UE policies in the response, encoded within an MBIM\_TLV\_IE structure with the type of MBIM\_TLV\_TYPE\_UE\_POLICIES.

There may be optional, unnamed IE’s in MBIM\_TLV\_IE format after the first two named fields in MBIM\_MS\_NW\_PARAMS\_QUERY\_INFO. If *InformationBufferLength* implies there is remaining data in the payload, the modem shall attempt parse and process unnamed IE’s in the MBIM\_TLV\_IE payload, following the rules outlined in Section 3.2.1.

### RESPONSE

The following structure shall be used in the *InformationBuffer* of a query response. The *InformationBufferLength* is variable. There is no valid set response.

Table 3.8‑7: MBIM\_MS\_NW\_PARAMS\_ INFO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset | Size | Field | Type | Description |
| 0 | 4 | MicoInd | MBIM\_MS\_MICO\_IND | MICO Indication Included only when AreConfigurationsNeeded is set |
| 4 | 4 | DrxParams | MBIM\_MS\_DRX\_PARAMS | Negotiated DRX parameters Included only when AreConfigurationsNeeded is set |
| 8 | variable | Unnamed IE’s | octets | This payload may contain 0 or more unnamed and optional Information Elements (IE) encoded in MBIM\_TLV\_IE. Refer to 3.2 for the format and processing rules. Optional unnamed IE’s in MBIM\_TLV\_IE format. |

The response to a Configuration Group query will vary depending on which parameters have been received from the network. If AreConfigurationsNeeded is set in the query, the *InformationBuffer* of a successful response shall contain the MICO and DRX fields and any received Network Configuration group TLVs (see Table 3.8‑1).

If the corresponding query request indicates that the network parameters of the UE Policies group are needed, the *InformationBuffer* of a successful response must include an unnamed MBIM\_TLV\_IE IE with the type of MBIM\_TLV\_TYPE\_UE\_POLICIES and the TLV shall return the current UE policies for the SUPI associated with the current SIM.

If there is no SIM inserted at the time or if the SUPI associated with the current SIM cannot be obtained, the request shall fail with a Status code indicating the reason:

* MBIM\_STATUS\_SIM\_NOT\_INSERTED if SIM is not inserted.
* MBIM\_STATUS\_PIN\_REQUIRED if the SIM is locked.
* MBIM\_STATUS\_BAD\_SIM for other reasons where SIM is inserted but SUPI cannot be obtained.

There may be optional, unnamed IEs in MBIM\_TLV\_IE format after the first two named fields in MBIM\_MS\_NW\_PARAMS\_QUERY\_INFO. If *InformationBufferLength* implies there is remaining data in the payload, the modem shall attempt parse and process unnamed IEs in MBIM\_TLV\_IE in the payload, following the rules in Section 3.2.1.

### NOTIFICATION

The structure MBIM\_MS\_NW\_PARAMS\_ INFO shall be used in the *InformationBuffer* of a notification. The *InformationBufferLength* is variable.

Whenever a network parameter in either network parameters group is changed in the modem by network, the modem shall send a notification of this CID.

If any parameter in the group of Network-set UE policies is changed, such as a policy section added, removed or modified, the notification shall include an MBIM\_TLV\_IE structure with the type of MBIM\_TLV\_TYPE\_UE\_POLICIES and the TLV shall return the current UE policies for the SUPI associated with the current SIM.

If no parameter in the group of Network-set UE policies is changed, the notification shall not include an MBIM\_TLV\_IE structure with the type of MBIM\_TLV\_TYPE\_UE\_POLICIES.

If any parameter in the Network Configurations group is changed, the notification shall include the corresponding Network Configuration Information Element as defined Table 3.1‑1.

### STATUS CODES

This CID uses Generic Status Codes (see Use of the Status Codes section 9.4.5 of [1]).

# Appendix

## Common Data Structures and Definitions

This section defines the commonly used data structures.

Table 4.1‑1: MBIM\_MS\_DNN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 1 | 1 | DnnLength | UINT8 | DNN Length. Minimum length of 3 octets and a maximum length of 102 octets |
| 2 | Variable | Dnn | CHAR | Data Network Name |

Table 4.1‑2: MBIM\_MS\_PLMN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 2 | Mcc | UINT16 | Mobile Country Code of 3 decimal digits; The least significant 12 bits contains BCD-encoded 3 decimal digits sequentially for the MCC, with the last digit of the MCC in the least significant 4 bits. The unused bits in the UINT16 integer must be zeros. |
| 2 | 2 | Mnc | UINT16 | Mobile Network Code of either 3 or 2 decimal digits; The most significant bit indicates whether the MNC has 2 decimal digits or 3 decimal digits. If this bit has 1, the MNC has 2 decimal digits and the least significant 8 bits contains them in BCD-encoded form sequentially, with the last digit of the MNC in the least significant 4 bits. If the most significant bit has 0, the MNC has 3 decimal digits and the least significant 12 bits contains them in BCD-encoded form sequentially, with the last digit of the MNC in the least significant 4 bits. The unused bits in the UINT16 integer must be zeros. |

Table 4.1‑3 defines the format for a single Tracking Area Identity (TAI). It consists of a PLMN and a Tracking Area Code (TAC).

Table 4.1‑3: MBIM\_MS\_TAI

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 4 | Plmn | MBIM\_MS\_PLMN | PLMN (MCC + MNC) Table 4.1‑2: MBIM\_MS\_PLMN. All zeros in the MCC component of this field indicates unknown TAI. Note that 000 is not a valid and defined MCC. |
| 4 | 4 | Tac | UINT32 | Tracking Area Code. If TAC is unknown or invalid, entire TAI is unknown and the MCC component of the field above should be set as mentioned above. |

Table 4.1‑4 defines the format for a TAC list belonging to a single PLMN – suitable for consecutive and non-consecutive TAC values.

Table 4.1‑4: MBIM\_MS\_TAI\_LIST\_SINGLE\_PLMN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 4 | Plmn | MBIM\_MS\_PLMN | PLMN (MCC + MNC) Table 4.1‑2: MBIM\_MS\_PLMN |
| 4 | 1 | ElementCount | UINT8 | Number of elements in the TAC list. Valid range is 1 to 16. |
| 5 | Variable | TacList[] | UINT32[] | Tracking Area Code (TAC) List The number of elements is specified by the ElementCount. |

Table 4.1‑5 defines the format for a list of TAIs belonging to different PLMNs.

Table 4.1‑5: MBIM\_MS\_TAI\_LIST\_MULTI\_PLMNS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Offset in octets | Size in octets | Field | Type | Description |
| 0 | 1 | ElementCount | UINT8 | Number of elements in the TAI list. Valid range is 1 to 16. |
| 1 | Variable | TaiList[] | MBIM\_MS\_TAI[] | Tracking Area Identity (TAI) List Table 4.1‑3 The number of elements is specified by the ElementCount. |

Table 4.1‑6: MBIM\_MS\_DRX\_PARAMS

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMDRXNotSpecified | 0 | DRX cycle not specified. In set requests, this indicates that the host does not request modem to use any DRX cycle. In set or query responses, this indicates that the DRX cycle to be used is unknown to the device at MBIM interface. |
| MBIMDRXNotSupported | 1 | The modem does not support setting DRX cycle at MBIM interface. |
| MBIMDRXCycle32 | 2 | DRX cycle T=32 |
| MBIMDRXCycle64 | 3 | DRX cycle T=64 |
| MBIMDRXCycle128 | 4 | DRX cycle T=128 |
| MBIMDRXCycle256 | 5 | DRX cycle T=256 |

Table 4.1‑7: MBIM\_MS\_ACCESS\_TYPE

|  |  |  |
| --- | --- | --- |
| Types | Value | Description |
| MBIMAccessTypeUnknown | 0 | AccessType is not known |
| MBIMAccessType3GPP | 1 | AccessType is 3GPP |
| MBIMAccessTypeNon3GPP | 2 | AccessType is Non-3GPP |

## Informative Sequence Diagrams

The section contains informative sequence diagrams. They illustrate how the MBIM extensions are used in typical scenarios between an MBIM host and an MBIM device.

### Initial Registration

When a UE is powered up and support 5G-NGC, the host will provide registration parameters to the MBIM device before the initial registration can be performed:



Figure 5: Initial Registration During Power On

1. The host queries the MBIM device capabilities by sending MBIM\_CID\_MS\_DEVICE\_CAPS\_INFO\_V2 query to the MBIM device. The MBIM device shall power up with radio state OFF.
2. The MBIM device reports its capabilities including supported data classes.
3. The host will query the current SIM state by sending MBIM\_CID\_SUBSCRIBER\_READY\_STATUS query
4. The MBIM device response its current SIM status with MBIM\_CID\_SUBSCRIBER\_READY\_STATUS response. If the MBIM device has already initialize its SIM, it will return readystate as “Initialized” and all other fields are valid and final. In case where e.g. SUPI/IMSI or telephone number(s) were not yet available for MBIM device, it shall return readystate as “NotInitialized”
5. If previous CID (4) indicated that readystate was “NotInitilized”, the MBIM device shall send unsolicited MBIM\_CID\_SUBSCRIBER\_READY\_STATUS notification when SIM has been initialized and all fields are valid.
6. The host will only then query the modem config status by sending MBIM\_CID\_MS\_MODEM\_CONFIG query to the MBIM device.
7. If the MBIM device has already completed the configuration, it shall response with ConfigStatus as “Completed”, otherwise, it shall report “Started” if configuration is pending or “Unknown” if the configuration has not yet started.
8. If the previous CID (7) ConfigStatus was either “Started” or “Unknown”, the host will expect an unsolicited MBIM\_CID\_MS\_MODEM\_CONFIG notification with ConfigStatus as “Started” or “Completed”. In case of “Completed” the MBIM device also assign configuration ID and preferred NSSAI if the configuration holds one.
9. The host configures the registration parameters to the MBIM device. If the modem reported “preferred NSSAI” in (7/8), or there is no “preferred NSSAI” configured either side, then the host will not send “preferred NSSAI”.
10. The MBIM device responses back status indication success. There is no retry in case of failure.
11. Once the MBIM device has the registration parameters configured, the host will turn the software radio ON allowing the MBIM device to camp to a cell. It is assumed that the MBIM device always powers on with radio OFF state.
12. If the MBIM device camps on a 5G-NGC cell, it will send an initial registration to the network. The parameters configured by the host in (9) shall be used, but the MBIM device is responsible for the rest of the parameters including UPSI and SMS NAS.
13. After the registration process is successfully completed, the network will send REGISTRATION\_ACCEPT response with Allowed/Configured/Rejected NSSAI lists, MICO status, LADN Information, etc.
14. The MBIM device informs the host that registration is successfully completed by sending unsolicited MBIM\_CID\_REGISTRATION\_STATE notification followed by
15. MBIM\_CID\_PACKET\_SERVICE having a subclass indicating the current network configuration and current Tracking Area Identity of a current cell.
16. Optionally, the MBIM device can send network specific registration parameters only after (10) and (11) by sending unsolicited MBIM\_CID\_MS\_NETWORK\_PARAMS notification if it has received them from the network. Otherwise, the host will query them from the MBIM device. In MBIMEx 3.0, the CID MBIM\_CID\_MS\_NETWORK\_PARAMS is optional. This step happens only if the device supports it. The host operating in MBIMEx 3.0 only uses URSP rules from this CID for informational purpose.

### Initial URSP Rules Retrieval

Modem is responsible of persistently store URSPs in non-volatile memory, along with the SUPI associated with the rules. URSP rules can only be used when the SUPI of the current SIM matches the SUPI associated with the URSP rules. The diagram below shows the activities and interaction for retrieval of stored URSP rules for a SIM.

In MBIMEx 3.0, the CID MBIM\_CID\_MS\_NETWORK\_PARAMS is optional. This scenario applies only if the device supports it. The host operating in MBIMEx 3.0 only uses URSP rules for informational purpose.



Figure 6: Initial URSP Rules Retrieval

### PDU Session Establishment with IPv4v6

The diagram below shows the activaties and interactions for a PDU session establishment for an Ipv4v6 session type.



Figure 7: PDU Session Establishment for IPv4v6 type

### URSP Rules Update

After registration, the HPLMN may update the UE policies, including URSP rules, in a UE. The network uses the MANAGE UE POLICIES procedure to do that. The diagram in Figure 8 shows the interactions between an MBIM host and an MBIM device for URSP rules update.

In MBIMEx 3.0, the CID MBIM\_CID\_MS\_NETWORK\_PARAMS is optional. This scenario applies only if the device supports it. The host operating in MBIMEx 3.0 only uses URSP rules for informational purpose.



Figure 8: URSP Rules Update by NGC

### Registration after SIM Hot-swap

When a SIM card is swapped when a device is powered up, configuration for the 5G-NGC registration shall be done before the initial registration is sent to the 5G Core as shown in Figure 9.



Figure 9: Registration after SIM Hot-Swap

1. The MBIM device sends unsolicited MBIM\_CID\_SUBSCRIBER\_READY\_STATUS notification when SIM is removed with ReadyState set as “SIMNotInserted”.
2. When the MBIM device detects a new SIM, it will send unsolicited MBIM\_CID\_SUBSCRIBER\_READY\_STATUS notification with ReadyState set as “NotInitialized”.
3. The host will turn the software radio OFF to delay the MBIM device for registration before the registration parameters are passed to the MBIM device.
4. The MBIM device will send unsolicited MBIM\_CID\_SUBSCRIBER\_READY\_STATUS notification when SIM has been initialized with ReadyState set as “Initialized”.
5. The MBIM starts to load the modem configuration and sends unsolicited MBIM\_CID\_MS\_MODEM\_CONFIG\_STATUS notification where ConfigStatus field is set to “Started”.
6. After the loading of a configuration is completed, the MBIM device sends unsolicited MBIM\_CID\_MS\_MODEM\_CONFIG notification, where ConfigStatus field is set to “Completed” The MBIM device also assign configuration ID and preferred NSSAI if the configuration holds one.
7. The host configures the registration parameters to the MBIM device. If the modem reported “preferred NSSAI” in (3), or there is no “preferred NSSAI” configured either side, then the host will not send “preferred NSSAI”.
8. The MBIM device responses back status indication success. There is no retry in case of failure.
9. The host turn the software radio ON to allow the MBIM device to camp to a cell and register.
10. Once the MBIM device has camped to a cell, it sends registration request to the 5G Core. The parameters configured by the host in (5) shall be used, but the MBIM device is responsible for the rest of the params like UPSI and SMS NAS.
11. After the registration process is successfully completed, the network will send REGISTRATION\_ACCEPT response with Allowed/Configured/Rejected NSSAI lists, MICO status, LADN Information etc.
12. The MBIM device informs the host that registration is successfully completed by sending unsolicited MBIM\_CID\_REGISTRATION\_STATE notification followed by
13. MBIM\_CID\_PACKET\_SERVICE having a subclass indicating the current network configuration and current Tracking Area Identity of a current cell.
14. Optionally, the MBIM device can send network specific registration parameters only after (9) and (10) by sending unsolicited MBIM\_CID\_MS\_NETWORK\_PARAMS notification if it has received them from the network. Otherwise, the host will query them from the MBIM device. In MBIMEx 3.0, the CID MBIM\_CID\_MS\_NETWORK\_PARAMS is optional. This step happens only if the device supports it. The host operating in MBIMEx 3.0 only uses URSP rules from this CID for informational purpose.

### Detection of SIM Swap while hibernation and URSP Rules Retrieval

After a system is fully functioning with a SIM, it may go into hibernation. While in hibernation, the modem may stay in bus, but the host and the modem stop communication. During hibernation, the user may remove the SIM from the system and inserts a new SIM.

After the system resumes from hibernation, the modem runs independent of, and often ahead of the host. Figure 10 shows the activities and interaction for such a situation. The host detects that SIM was swapped while the system was in hibernation. A difference from SIM hot swap is that the host does not get an explicit notification about SIM removal in this case.



Figure 10: Detection of SIM Swap during Hibernation and URSP Rules Retrieval

### Registration after SIM Swap in Hibernate Mode

When a SIM card is swapped when a device is in hibernate state, during resume from hibernate, a configuration for the 5G-NGC registration shall be done before the initial registration is sent to the 5G Core:



Figure 9 Registration after SIM Swap in Hibernate Mode

1. The device is in Hibernate mode and a user switches to a new SIM.
2. The host resumes from the hibernate mode and sends MBIM\_SUBSCRIBER\_REASY\_STATYS query to the MBIM device.
3. The MBIM device responds with the latest SIM status. Depending of the MBIM device state, the following options are possible:
   1. If the SIM is NotInitialized state, the host will continue the flow as in (2) of Figure5.
   2. If the SIM is already initialized, and the MBIM device has completed the configuration, it shall not send any unsolicited MBIM\_CID\_MS\_MODEM\_CONFIG notifications. The host will send a query to request of the current status of configuration.
   3. If the SIM is already initialized, but the MBIM device is in middle of the configuration or not yet started, it shall send unsolicited MBIM\_CID\_MS\_MODEM\_CONFIG notifications with config status as “Started” or “Unknown” followed by “Completed”. In this case, the flow continues as in (5) of Figure 5.
4. In case of (3b), where the MBIM device already loaded new configuration, it may start the initial registration.
5. The host will query the modem configuration state by sending unsolicited MBIM\_CID\_MS\_MODEM\_CONFIG notification if ReadyState in (3) was set as “Initialized”
6. The MBIM device shall responsd with ConfigState as “Unknown”. In this case, the flow continues as (4) in Figure 5. If the config status is “Started”, then the flow continues as (5) in Figure 5.
   1. If the MBIM device sent the initial registration in (4), unsolicited (8), (9) and optional (10) can be sent to the host at any time.
7. The host configures the registration parameters to the MBIM device. If the modem has reported “preferred NSSAI” in (6), or there is no “preferred NSSAI” configured either side, then the OS will not send “preferred NSSAI”.
8. The MBIM device responses back status indication success. There is no retry in case of failure.
9. If any registration params given in (11) are different than currently stored in the MBIM device, the MBIM device shall trigger the re-registration. The parameters configured by the host in (11) shall be used, but the MBIM device is responsible for the rest of the params like UPSI and SMS NAS.
10. After the registration process is successfully completed, the network will send REGISTRATION\_ACCEPT response with Allowed/Configured/Rejected NSSAI lists, MICO status, LADN Information etc.
11. The MBIM device informs the host that registration is successfully completed by sending unsolicited MBIM\_CID\_REGISTRATION\_STATE notification followed by
12. MBIM\_CID\_PACKET\_SERVICE having a subclass indicating the current network configuration and current Tracking Area Identity of a current cell.
13. Optionally, he MBIM device can send network specific registration parameters only after (9) and (10) by sending unsolicited MBIM\_CID\_MS\_NETWORK\_PARAMS notification if it has received them from the network. Otherwise, the host will query them from the MBIM device. In MBIMEx 3.0, the CID MBIM\_CID\_MS\_NETWORK\_PARAMS is optional. This step happens only if the device supports it. The host operating in MBIMEx 3.0 only uses URSP rules from this CID for informational purpose.

## Successful Secondary Authentication

The diagram below shows the activaties and interactions for a secondary authentication during PDU session establishment for an Ipv4v6 session type.



Figure10: Secondary Authentication during PDU session establishment

## Clarification on Status Code MBIM\_STATUS\_BUSY

Unless a CID explicitly defines the purpose and meaning of this status code in responses in its respective specification, the usage of status code MBIM\_STATUS\_BUSY in responses shall conform to the following specification for MBIMEx version 3.

A status code MBIM\_STATUS\_BUSY in a response signifies that the corresponding request has failed due to a busy condition in the device and that the request session is closed. After a failure response with this status code is sent, the modem shall not send any further response to the original request and any such further response shall be ignored by the host.

This is for the context of MBIMEx version 3.0 when it is the governing MBIM Extensions Release number between an MBIM host and MBIM device. It applies to new and modified CIDs in MBIMEx version 3.0, as well as the inherited and otherwise unmodified CIDs from MBIMEx versions 1.0 and 2.0.

This does not apply to notifications.

## List of General Improvements

Throughout this document, general improvements, enhancements and/or clarifications are made in MBIMEx version 3 that applies to both 5GC-capable devices and 5GC-incapable devices. The table below summarizes those general improvements, enhancements and/or clarifications, the CIDs that are affected, and the section where they are made.

|  |  |  |
| --- | --- | --- |
| Enhancements | Relevant CID(s) | Where they are made |
| Enhancement and clarification on how status code MBIM\_STATUS\_BUSY is used in responses | All CIDs that may return the status of MBIM\_STATUS\_BUSY in MBIMEx version 3.0 | Section 4.4 |
| Enhancement and clarification on ready state of  MBIMSubscriberReadyStateInitialized | MBIM\_CID\_SUBSCRIBER\_READY\_STATUS in MBIMEx version 3.0 | Section 3.4.1 |
| Enhancement and clarification on the usage of ready state of  MBIMSubscriberReadyStateUninitialized | MBIM\_CID\_SUBSCRIBER\_READY\_STATUS in MBIMEx version 3.0 | Section 3.4.1 |
| Clarification on host’s responsibility of detecting SIM swap during hibernation. | MBIM\_CID\_SUBSCRIBER\_READY\_STATUS in MBIMEx version 3.0 | Section 3.4.1 |
| Enhancement on reporting for eSIM and SIM form factor | MBIM\_CID\_SUBSCRIBER\_READY\_STATUS in MBIMEx version 3.0 | Sections 3.4.2, 3.4.4, and 3.4.5 |
| Clarification that modems must support SLAAC for IPv6 stateless address autoconfiguration if an activated context/session contains an IPv6 portion | MBIM\_CID\_CONNECT in MBIMEx version 3.0 | Section 3.9.1 |
| Compatibility of requested session types vs activated session types | MBIM\_CID\_CONNECT in MBIMEx version 3.0 | Section 3.9.4 |
| Enhancement on reporting WCDMA/LTE/NR Radio Bands | MBIM\_CID\_DEVICE\_CAPS\_V2 in MBIMEx version 3.0 | Section 3.3.3 |
| Enhancement on report device capabilities | MBIM\_CID\_DEVICE\_CAPS\_V2 in MBIMEx version 3.0 | Sections 3.3.2 and 3.3.3 |

# References

1. USB Forum, “Universal Serial Bus Communications Class Subclass Specification for Mobile Broadband Interface Model,” revision 1.0 Errata-1, 1 May 2013, <http://www.usb.org/developers/docs/devclass_docs/MBIM10Errata1_073013.zip>
2. MSDN: [MBIM\_CID\_MS\_DEVICE\_CAPS\_V2](https://docs.microsoft.com/en-us/windows-hardware/drivers/network/mb-multi-sim-operations)
3. 3GPP TS 23.501: "System architecture for the 5G System (5GS)"
4. 3GPP TS 23.502: "Procedures for the 5G System (5GS)"
5. 3GPP TS 24.501: "Non-Access-Stratum (NAS) for the 5G System (5GS); Stage 3"
6. 3GPP TS 24.526: “UE policies for 5G Systems (5GS); Stage 3”
7. 3GPP TS 23.003: “Numbering, addressing and identification”
8. 3GPP TS 23.503: “Policy and Charging Control Framework for 5G System; Stage 2”
9. RFC 4862, “IPv6 Stateless Address Autoconfiguration”, Sept 2007
10. 3GPP TS 38.133, “5G; NR; Requirements for support of radio resource management”
11. 3GPP TS 38.331, “5G; NR; Radio Resource Control (RRC)”
12. MSDN: [MBIM\_CID\_BASE\_STATIONS\_INFO](https://docs.microsoft.com/en-us/windows-hardware/drivers/network/mb-base-stations-information-query-support)
13. GSAM: “[Road to 5G: Introduction and Migration](https://www.gsma.com/futurenetworks/wp-content/uploads/2018/04/Road-to-5G-Introduction-and-Migration_FINAL.pdf)”, April 2018