Qualcomm Technologies, Inc.

Qualcomm Aware™ Positioning Service XML API

Reference Manual

80-42216-1 Rev. AB

October 15, 2024

Revision history

Revision	Date	Description
AB	October 2024	■ Updated the product name from Qualcomm Terrestrial Positioning Service (TPS) to Qualcomm Aware™ Positioning Service in the entire document. ■ Updated the document to conform to latest documentation standards. No technical content was changed in this revision.
AA	June 2022	Initial release

Contents

Tables

1 Introduction to Qualcomm Aware™ Positioning Service API

Qualcomm Aware™ Positioning Service API, formerly known as Skyhook Precision Location API, provides access to positioning information derived from analyzing positioning anchors such as Wi-Fi access points and cell IDs in known locations.

Client applications make XML HTTPS requests to the positioning service API, which include a list of observed beacons used for positioning within the range of the client device, and if available, GPS information starting from positioning service v2.21. Based on these inputs, the positioning service API returns a calculated geographic location, and optionally returns a street address and time zone.

In addition to the derived location, the positioning service API returns location information based on the IP address of the client. While this information is not a reliable source for navigation, it can be used to provide context when more accurate methods are not available, and may be used as the best estimate of location.

API limitations

Opting for the API approach has benefits pertaining to server-side features, but is less robust than client-side implementations. Few SDK or embedded client features such as smoothing between location sources, fast time-to-fix, MAC address deduplication, power optimization, and offline location are not available with the API method.

In addition, application or device-specific code must be created to enable the collection of Wi-Fi and cell radio scans. For recommendations in these areas to optimize the use of cloud API, contact the Qualcomm Aware™ Positioning Service team at support.tps@qti.qualcomm.com.

Scope

This document provides an overview of the positioning service XML API and its implementation to determine the geographic location of connected devices. In addition, the document describes the API requests and responses, and provides guidelines to evaluate the API on a large scale.

Technical assistance

For assistance or clarification, write to support.tps@qti.qualcomm.com.

2 Positioning service XML API implementation

To implement the positioning service API, the device must collect data about the radio signal environment, including Wi-Fi access point MAC addresses, cell tower IDs, and signal strengths. This data is then submitted using the positioning service API. Along with this data, if accessible, include GPS location information to improve location results even when GPS is unavailable. Starting from positioning service v2.21, GPS switching has been added to server-side functionality.

HTTP headers

For API calls, always use DNS instead of hard coded IP addresses as they are not supported by positioning service and can cause a failure in returning location requests.

If all API calls are forwarded from a centralized server before reaching the positioning service API endpoint, the x-forwarded-for HTTP header must be used with the original public API address of the device making the request.

Consider the following example of an HTTP header:

```
Forwarded: for=192.0.2.60; proto=http; by=203.0.113.43
```

A new optional `Skyhook-Request-Token` HTTP header has been introduced in protocol v2.28. Clients can choose to include an opaque `messageID` in the `Skyhook-Request-Token` header in any type of request, and the API server will echo back the provided value as a `Skyhook-Request-Token` header in the corresponding response.

API key

An evaluation license agreement is required to obtain an API key for the positioning service. For more information on the license agreement, contact the Qualcomm Aware™ Positioning Service team at sales.tps@qti.qualcomm.com.

Unique device ID

With each unique location request, it is recommended to submit a unique device ID or `username` for the end-user device. This enables more accurate location results and impacts the redundancy of how location data is processed and weighted for optimization. If `username` is not provided, reporting and API performance may be impacted.

NOTE Individual device IDs are not stored permanently in Qualcomm's system, but are stored with a rotating hash in accordance with Qualcomm user and commercial privacy policies.

Endpoint URL

The positioning service API endpoint URL of XML-formatted requests is https://global.skyhook.com/wps2/location.

2.1 HTTP authentication methods

Positioning service XML API request can be authenticated using:

- HTTP headers via:
  - Skyhook authentication key and username
  - Skyhook web token
- HTTP URL parameters
- XML body

HTTP header-based authentication

- **Skyhook auth-key and username**: Submit a request with `Skyhook-Auth-Key` header containing the API key along with an optional `Skyhook-Auth-User` containing the username associated with the key.

  NOTE An evaluation license agreement is required to obtain an API key for the positioning service. For more information on the license agreement, contact the Qualcomm Aware™ Positioning Service team at sales.tps@qti.qualcomm.com.

- **Skyhook web token**: Skyhook supports a token-based registration and authentication model, available on a per-request basis. For more information, contact the Qualcomm Aware™ Positioning Service team at support.tps@qti.qualcomm.com.

HTTP URL parameters-based authentication

The positioning service XML API request body does not contain an explicit authentication section. Rather, the API key and user ID are encoded into the request endpoint URL.

```
https://global.skyhookwireless.com/wps2/location?
key="<somekey>&username=<someusername>"
```

XML body-based authentication

NOTE Starting from positioning service v2.34, XML body-based authentication is deprecated. Previous versions of positioning service can still use this authentication method.

Use the following sample XML body for authentication:

```
<authentication version="2.2">
        <key key="YOUR API KEY HERE"
        username="DEVICE SERIAL NUM, MAC ADDRESS, OR OTHER UNIQUE ID HERE"/>
</authentication>
```

Where:

- `key` is the API key used for this project.
- `username` is the serial number, MAC address, or other unique ID of the device.
- `version` is the version of the authentication type used for this project.

3 Positioning service XML API request

Requests the geographic location of the device based on observed access points, cell towers, and GPS locations.

Endpoint URL

The positioning service API endpoint URL of XML-formatted requests is https://global.skyhook.com/wps2/location.

HTTP method

POST

HTTP authent cat on

See HTTP authentication methods.

Syntax

```
<LocationRQ xmlns="http://skyhookwireless.com/wps/2005"
            version="2.31"
            hpe-confidence="68"
            street-address-lookup="full"
            timezone-lookup="true"
             emergency="true>
<authentication version="2.2">
    ...
</authentication>
    <access-point> <!-- See below for details --> </access-point>
    <lte-tower> <!-- Any type of cell can be provided between APs and GPS -->
</lte-tower>
          <!-- See Scanned Cell Tower Group below for details -->
    <nr-tower> <!-- | --> </nr-tower>
          <!-- | -->
    <gps-location>  <!-- See below for details --> </gps-location>
    <eddystone-uid-ble>  <!-- Any type of BLE can be provided after GPS --> </
eddystone-uid-ble>
          <!-- See Scanned BLE Beacon Group below for details -->
       <altbeacon-ble>    <!-- | --> </altbeacon-ble>
          <!-- | -->
</LocationRQ>
```

Parameters

Parameter	Values	Description
`rqtime`	–	(Optional) The time at which the request data is scanned. This attribute is reflected back in the response.
`timezone-lookup`	▪ True ▪ False (default)	Requests time zone lookup in addition to lat/long lookup. ▪ Set to `false` to not perform time zone lookup. ▪ Set to `true` to perform time zone lookup, which yields a `timezone` in the `LocationRS`.
`street-address-lookup`	▪ None (default) ▪ Limited ▪ Full	Requests street address lookup in addition to lat/long lookup. ▪ Set to `none` to not perform street address lookup. ▪ Set to `limited` to return information of the city, which yields a `street-address` in the `LocationRS`. ▪ Set to `full` to return all address attributes, which yield a `street-address` in the `LocationRS`.
`version`	–	The version of the API used.
`hpeConfidence`	▪ 68 (default) ▪ 95	(Optional) Requests reported uncertainty (HPE) percentile change. For example, a value of 95 reports 95% uncertainty. **NOTE** The default 68% confidence is supported in positioning service v2.28 and later. For 95% confidence, set `hpe-confidence` to 95 in the location request.
`authentication`	–	The authentication for this API request. ▪ `key`: The API key used for this project. ▪ `username`: The serial number, MAC address, or other unique ID of the device. ▪ `version`: The version of the authentication type used for this project.
access-point	–	(Optional) An access point observed during a scan. Any number of access points can be provided in a request.
Scanned cell tower group	–	(Optional) A cell tower observed during a scan. Any number of cell towers can be provided in a request. The types of cell towers supported are as follows: ▪ gsm-tower ▪ umts-tower ▪ lte-tower (4G, LTE, CAT-M1, CAT-M2, LTE-M) ▪ cdma-tower ▪ nbiot-tower (CAT-NB1, CAT-NB2) ▪ nr-tower (5G, New Radio)
GPS location	–	(Optional) A GPS point observed during a scan. Any number of GPS points can be provided in a request.
Scanned BLE beacon group	–	(Optional) A BLE beacon observed during a scan. Any number of BLE beacons can be provided in a request. The types of BLE beacons supported are as follows: ▪ IBeacon BLE ▪ Eddystone BLE ▪ AltBeacon BLE

3.1 access-point

Provides the access point observed during a scan.

Syntax

```
<access-point>
    <mac>E01C413B9414</mac>
    <ssid>SkyFi-Corp</ssid>                <!-- Optional -->
    <signal-strength>-66</signal-strength>
    <age>100</age>                         <!-- Optional -->
    <freq>5260</freq>                      <!-- Optional -->
    <connected>true</connected>            <!-- Optional -->
</access-point>
```

Parameters

Parameter	Values	Description
`mac`	–	MAC address of the access point. MAC address must be complete and valid. Each character should be capitalized, and separators such as colons and dashes should be removed. For example: ■ Correct MAC address: 000C4182D88C ■ Incorrect MAC address: 00:0C:41:82:D8:8C
`ssid`	–	(Optional) Service Set Identifier (SSID) of the access point.
`signal-strength`	–	(Optional) Observed signal strength of the access point, in dBm.
`age`	–	(Optional) Relative age of the measurement, in milliseconds or timestamp.
`freq`	–	(Optional) Central channel frequency in MHz; 16-bit integer.
`connected`	■ True ■ False ■ Empty (default)	(Optional) Indicates whether the client device is currently connected to the access point. ■ `true`: connected ■ `false`: not connected ■ `empty`: not connected

3.2 Scanned cell tower group

The scanned cell tower group allows the input of observed cell towers. Multiple towers and multiple types of cell towers are allowed.

Optionally, the cell tower list can include neighbors cell measurements (NMRs) in addition to serving cells. Neighbor cells can be provided with the full cell IDs (as serving cell) or partial IDs (labeled in the table as NMRs).

NOTE Partial cell ID must be accompanied by at least one serving cell.

3.2.1 gsm-tower

Provides the GSM cell tower observed during a scan. The GSM tower block should be used for the following air-interfaces:

- GSM
- GPRS
- EDGE

Syntax

```
<gsm-tower>
    <mcc>310</mcc>
    <mnc>410</mnc>
    <lac>1234</lac>
    <ci>12345</ci>
    <bsic>35</bsic>                      <!-- Optional -->
    <rssi>-90</rssi>                     <!-- Optional -->
    <timing-advance>11</timing-advance>  <!-- Optional -->
    <age>1000</age>                      <!-- Optional -->
    <arfcn>330</arfcn>                   <!-- Optional -->
    <band>GSM_480</band>                 <!-- Optional -->
    <serving>false</serving>             <!-- Optional -->
</gsm-tower>
```

Parameters

Parameter	Values	Description
`mcc`	–	Indicates the mobile country code.
`mnc`	–	Indicates the mobile network code.
`lac`	–	Indicates the local area code.
`ci`	–	Indicates the cell ID.
`bsic`	Range: 0 to 63	(Optional) GSM BSIC composed of a 3-bit Network Color Code (NCC) and a 3-bit Base station Color Code (BCC).
`rssi`	–	Signal strength of the cell, in dBm.
`timing-advance`	Range: 0 to 63	(Optional) GSM time advance value (TA) when in connected mode. Integer value as per 3GPP Technical Specification 45.00 (GSM).
`age`	–	Relative age of the measurement, in milliseconds or timestamp.
`arfcn`	–	(Optional) GSM channel number; 16-bit integer.
`band`	–	(Optional) GSM band; string.
`serving`	▪ True ▪ False ▪ Empty (default)	(Optional) Indicates whether the client device is currently associated with the cell tower. ▪ `true`: connected/serving cell ▪ `false`: not a serving cell ▪ `empty`: not serving

3.2.2 umts-tower

Provides the UMTS cell tower observed during a scan. The UMTS tower block should be used for the following air-interfaces:

- WCDMA
- UMTS-FDD
- UMST-TDD
- UTRAN
- TD-SCDMA
- HSPA

Syntax

```
<umts-tower>
    <mcc>310</mcc>
    <mnc>410</mnc>
    <lac>12345</lac>                    <!-- Optional -->
    <ci>1234567</ci>
    <psc>501</psc>                      <!-- Optional -->
    <rscp>-90</rscp>                    <!-- Optional -->
       <ec-no>-13</ec-no>                  <!-- Optional -->
       <age>100</age>                      <!-- Optional -->
    <uarfcn>1400</uarfcn>               <!-- Optional -->
    <serving>false</serving>            <!-- Optional -->
</umts-tower>
```

Parameters

Parameter	Values	Description
`mcc`	–	Indicates the mobile country code.
`mnc`	–	Indicates the mobile network code.
`lac`	–	Indicates the local area code.
`ci`	–	UMTS UC_ID (combination of RNDid and Cell_ID) with a total of 28 bits as per 3GPP Technical Specification 25.401. Populating cell ID with 16 bits may result in incorrect tower information.
`psc`	Range: 0 to 511	(Optional) UMTS local cell ID.
`rscp`	–	(Optional) Signal strength of the UMTS cell, in dBm.
`ec-no`	Range: -24 to 1	(Optional) Signal quality of the UMTS cell.
`age`	–	Relative age of the measurement in milliseconds or timestamp.

Parameter	Values	Description
`uarfcn`	–	(Optional) UMTS channel number; 16-bit integer.
`serving`	■ True ■ False ■ Empty (default)	(Optional) Indicates whether the client device is currently associated with the cell tower. ■ `true`: connected/serving cell ■ `false`: not a serving cell ■ `empty`: not serving

3.2.3 lte-tower (4G, LTE, CAT-M1, CAT-M2, LTE-M)

Provides the LTE cell tower observed during a scan. The LTE tower block should be used for the following air-interfaces:

- LTE
  - E-UTRAN
  - LTE-TDD
  - LTE-FDD

Syntax

```
<lte-tower>
    <mcc>310</mcc>
    <mnc>410</mnc>
    <tac>12345</tac>                          <!-- Optional -->
    <eucid>123456</eucid>
    <pci>123</pci>                            <!-- Optional -->
    <rsrp>-90</rsrp>                          <!-- Optional -->
       <rsrq>-29</rsrq>                       <!-- Optional -->
       <rssnr>11</rssnr>                         <!-- Optional -->
       <cqi>12</cqi>                             <!-- Optional -->
       <timing-advance>66</timing-advance>       <!-- Optional -->
    <age>100</age>                            <!-- Optional -->
    <earfcn>1105</earfcn>                     <!-- Optional -->
    <serving>false</serving>                  <!-- Optional -->
</lte-tower>
```

Parameters

Parameter	Values	Description
`mcc`	–	Indicates the mobile country code.
`mnc`	–	Indicates the mobile network code.
`tac`	–	(Optional) Indicates the tracking area code; 16-bit integer.
`eucid`	–	LTE E-CGI; 28 bits.
`pci`	Range: 0 to 503	(Optional) LTE local cell ID.

Parameter	Values	Description
`rsrp`	–	(Optional) Signal strength received from the LTE cell tower, in dBm.
`rsrq`	Range: -34 to 3	(Optional) Signal quality of the LTE cell.
`rssnr`	Range: -20 to 30	(Optional) Signal-to-noise ratio of the LTE cell.
`cqi`	Range: 0 to 15	(Optional) Channel quality indicator of the LTE cell.
`timing-advance`	■ Default range: 0 to 1282 ■ Extended range: 0 to 7690	(Optional) Length of time a signal takes to reach the base station from a mobile phone, half of round-trip time (RTT). The units are LTE 16xTs symbols as specified in Section 4.2.3 of 3GPP Technical Specification 36.213 and 36.133 (LTE).
`age`	–	Relative age of the measurement in milliseconds or timestamp.
`earfcn`	–	(Optional) 3GPP channel number; 18-bit integer.
`serving`	■ True ■ False ■ Empty (default)	(Optional) Indicates whether the client device is currently associated with the cell tower. ■ `true`: connected/serving cell ■ `false`: not a serving cell ■ `empty`: not serving

3.2.4 nbiot-tower (CAT-NB1, CAT-NB2)

Provides the NB-IoT cell tower observed during a scan. The NB-IoT tower block should be used for low-power IoT networks based on NB-IoT air-interfaces.

Syntax

```
<nbiot-tower>
    <mcc>310</mcc>
    <mnc>410</mnc>
    <tac>12345</tac>                    <!-- Optional -->
    <cid>123456</cid>
    <ncid>123</ncid>                    <!-- Optional -->
    <nrsrp>-90</nrsrp>                  <!-- Optional -->
    <age>100</age>                      <!-- Optional -->
    <earfcn>1105</earfcn>               <!-- Optional -->
    <serving>false</serving>            <!-- Optional -->
</nbiot-tower>
```

Parameters

Parameter	Values	Description
`mcc`	–	Indicates the mobile country code.
`mnc`	–	Indicates the mobile network code.
`tac`	–	(Optional) Indicates the NB-IoT tracking area code; 16 bit
`cid`	–	Indicates the NB-IoT cell ID; 28 bit.
`ncid`	Range: 0 to 503	(Optional) Indicates the NB-IoT local cell ID; equivalent to PCI in LTE.

Parameter	Values	Description
`nrsrp`	–	(Optional) Signal strength received from the NB-IoT tower, in dBm.
`age`	–	Relative age of the measurement, in milliseconds or timestamp.
`earfcn`	–	(Optional) 3GPP channel number; 16-bit integer.
`serving`	■ True ■ False ■ Empty (default)	(Optional) Indicates whether the client device is currently associated with the cell tower. ■ `true`: connected/serving cell ■ `false`: not a serving cell ■ `empty`: not serving

3.2.5 cdma-tower

Provides the CDMA cell tower observed during a scan. The CDMA tower block should be used for the following air-interfaces:

- CDMA
- CDMA2000
- EV-DO
- IS-95
- 1xRTT

Syntax

```
<cdma-tower>
    <sid>510</sid>
    <nid>89</nid>
    <bsid>1500</bsid>
    <pncode>31</pncode>                    <!-- Optional -->
    <cdma-lat>41.123456</cdma-lat>        <!-- Optional -->
    <cdma-lon>71.123456</cdma-lon>        <!-- Optional -->
    <pilot-power>-90</pilot-power>        <!-- Optional -->
    <age>100</age>                          <!-- Optional -->
    <channel-number>1105</channel-number>   <!-- Optional -->
    <band>1800</band>                       <!-- Optional -->
    <serving>false</serving>                <!-- Optional -->
</cdma-tower>
```

Parameters

Parameter	Values	Description
`sid`	–	Indicates the CDMA system identifier (SID).
`nid`	–	Indicates the CDMA network identifier (NID).
`bsid`	–	Indicates the CDMA base system identifier (BSID).
`pncode`	Range: 0 to 511	(Optional) Indicates the CDMA local cell ID.

Parameter	Values	Description
`timingAdvance`	Range: 0 to 7690	Length of time a signal takes to reach the base station from a mobile phone, half of round-trip time (RTT). The units are Ts symbols as specified in 3GPP Technical Specification 36.133 (LTE).
`pilot-power`	–	(Optional) Signal strength (pilot power) of the cell received from the tower, in dBm.
`age`	–	Relative age of the measurement, in milliseconds or timestamp.
`channel-number`	–	(Optional) CDMA channel number; 18 bit.
`band`	–	(Optional) CDMA band; string.
`serving`	■ False (default) ■ True	(Optional) Indicates whether the client device is currently associated with the cell tower. ■ `true`: connected/serving cell ■ `false`: not a serving cell ■ `empty`: not serving

Parameter	Values	Description
`sid`	–	Indicates the CDMA system identifier.
`nid`	–	Indicates the CDMA network identifier.
`bsid`	–	Indicates the CDMA base system identifier.
`pncode`	Range: 0 to 511	(Optional) Indicates the CDMA local cell ID.
`cdma-lat`	–	(Optional) The latitude of the CDMA tower.
`cdma-lon`	–	(Optional) The longitude of the CDMA tower.
`pilot-power`	–	(Optional) Signal strength received from the CDMA tower, in dBm.
`age`	–	Relative age of the measurement, in milliseconds or timestamp.
`channel-number`	–	(Optional) CDMA channel number; 16-bit integer.
`band`	–	(Optional) CDMA band; string.
`serving`	■ True ■ False ■ Empty (default)	(Optional) Indicates whether the client device is currently associated with the cell tower. ■ `true`: connected/serving cell ■ `false`: not a serving cell ■ `empty`: not serving

3.2.6 nr-tower (5G, New Radio)

Provides the 5G NR cell tower observed during a scan. The NR tower block should be used for cell towers in a 5G network.

Syntax

```
<nr-tower>
    <mcc>600</mcc>
```

```
    <mnc>10</mnc>
    <tac>25187</tac>                  <!-- Optional -->
    <nci>6871947673</nci>
    <pci>3400</pci>                   <!-- Optional -->
    <csi-rsrp>-50</csi-rsrp>                  <!-- Optional -->
    <rsrq>-10</rsrq>                          <!-- Optional -->
    <sinr>-11</sinr>                          <!-- Optional -->
    <cqi>-15</cqi>                            <!-- Optional -->
    <signal-ref-type>csi</signal-ref-type>    <!-- Optional -->
    <timing-advance>0</timing-advance>        <!-- Optional -->
    <age>0</age>                    <!-- Optional -->
    <nrarfcn>4000</nrarfcn>               <!-- Optional -->
    <serving>true</serving>               <!-- Optional -->
</nr-tower>
```

Parameters

Parameter	Values	Description
`mcc`	–	Indicates the mobile country code.
`mnc`	–	Indicates the mobile network code.
`tac`	–	(Optional) Indicates 16-bit NR tracking area code.
`nci`	–	Indicates the 5G local cell ID; 36 bits.
`pci`	Range: 0 to 1007	(Optional) Indicates the 5G local cell ID; equivalent to PCI in LTE.
`csi-rsrp`	–	(Optional) Signal strength received from the NR tower, in dBm.
`rsrq`	Range: -20 to 3	(Optional) Signal quality of the NR cell.
`sinr`	Range: -23 to 23	(Optional) Signal-to-noise ratio of the NR cell.
`cqi`	–	(Optional) Indicator of NR cell channel quality.
`signal-ref-type`	[csi, ss]	(Optional) Type of signal reference. Applicable to 5G NR cells. The values are case-sensitive; enum with values [csi, ss]. If this value is not available, use the measurements provided.
`timing-advance`	Range: 0 to 3846	(Optional) Length of time a signal takes to reach the base station from a mobile phone, half of round-trip time (RTT). The units are Ts as specified in 3GPP Technical Specification 28.212.
`age`	–	Relative age of the measurement, in milliseconds or timestamp.
`nrarfcn`	–	(Optional) 3GPP channel number; 24-bit integer.
`serving`	▪ True ▪ False ▪ Empty (default)	(Optional) Indicates whether the client device is currently associated with the cell tower. ▪ `true`: connected/serving cell ▪ `false`: not a serving cell ▪ `empty`: not serving

3.2.7 Neighbor cell reporting

Neighbor cells improve cell-based accuracy and yield. The positioning service XML API supports reporting of neighbor cells with partial identity.

The key requirements for neighbor cell reporting are as follows:

- At least one cell must be fully qualified, that is, it must include all nonoptional cell ID elements.
- Neighbor cells with partial identity should include at least local cell ID, such as `pci` in LTE, and channel frequency.
- When the serving cell is 4G (LTE), for example, it is common to see neighbor cells from 4G/3G/2G.
- When available, neighbor cells can be reported with full identity.
- Neighbor cells can be reported using channel frequency from the serving cell and intra-frequency flag. Such neighbor cells must be of the same cell type as the serving cell to use this feature.

Sample request

The following example includes a fully qualified serving LTE cell with all cell identifiers specified, an LTE neighbor cell, and a UMTS neighbor cell:

```
<lte-tower>                    <!-- Serving Cell -->
    <mcc>310</mcc>
    <mnc>410</mnc>
    <tac>12345</tac>
    <eucid>123456</eucid>
    <pci>123</pci>                      <!-- Optional -->
    <rsrp>-90</rsrp>                    <!-- Optional -->
    <age>100</age>                      <!-- Optional -->
    <earfcn>1105</earfcn>               <!-- Optional -->
    <serving>true</serving>             <!-- Optional -->
</lte-tower>
<lte-tower>                    <!-- LTE Neighbor Cell -->
    <pci>123</pci>
    <rsrp>-90</rsrp>                    <!-- Optional -->
    <age>1100</age>                     <!-- Optional -->
    <earfcn>1105</earfcn>
</lte-tower>
<lte-tower>           <!-- LTE intra-freq Neighbor Cell -->
    <pci>150</pci>
    <intra-freq>true</intra-freq>
</lte-tower>
<umts-tower>                   <!-- UMTS Neighbor Cell -->
    <psc>501</psc>
    <rscp>-90</rscp>                    <!-- Optional -->
    <age>500</age>                      <!-- Optional -->
    <uarfcn>1400</uarfcn>
</umts-tower>
```

```
<umts-tower>            <!-- UMTS intra-freq Neighbor Cell -->
    <psc>73</psc>
    <intra-freq>true</intra-freq>
</umts-tower>
<nbiot-tower>                   <!-- NBIOT Neighbor Cell -->
    <ncid>501</ncid>
    <nrscp>-90</nrscp>                  <!-- Optional -->
    <age>500</age>                      <!-- Optional -->
    <earfcn>1400</uarfcn>
</nbiot-tower>
<nbiot-tower>            <!-- NBIOT intra-freq Neighbor Cell -->
    <ncid>73</ncid>
    <intra-freq>true</intra-freq>
</nbiot-tower>
<nr-tower>                      <!-- NR (5G) Neighbor Cell -->
    <pci>501</pci>
    <csi-rscp>-90</csi-rscp>            <!-- Optional -->
    <age>500</age>                      <!-- Optional -->
    <nrarfcn>1400</uarfcn>
</nr-tower>
<nr-tower>            <!-- NR intra-freq Neighbor Cell -->
    <pci>73</pci>
    <intra-freq>true</intra-freq>
</nr-tower>
```

3.2.8 GPS location

Provides the GPS location observed during a scan.

Syntax

```
<gps-location fix="1">
    <latitude>11.123456</latitude>
    <longitude>11.123456</longitude>
    <hpe>19</hpe>
    <altitude>19.1</altitude>                       <!-- Optional -->
    <vpe>2</vpe>                                    <!-- Optional -->
    <speed>3.1</speed>                              <!-- Optional -->
    <bearing>60</bearing>                           <!-- Optional -->
    <age>1000</age>
    <speed-uncertainty>0.1</speed-uncertainty>      <!-- Optional -->
    <bearing-uncertainty>5</bearing-uncertainty>     <!-- Optional -->
    <constellation>
        <name>GPS</name>
        <nsat>4</nsat>
    </constellation>
    <constellation>
        <name>GLONASS</name>
```

```
        <nsat>5</nsat>
    </constellations>
    <constellation>
        <name>GALILEO</name>
        <nsat>3</nsat>
    </constellations>
</gps-location>
```

Parameters

Parameter	Values	Description
`fix`	■ 1 – GPS (default) ■ 2 – DGPS ■ 3 – PPS ■ 4 – RTK ■ 5 – FRTK ■ 6 – Estimated ■ 7 – MIM ■ 8 – SM	Type of fix.
`latitude`	–	Calculated physical geographic location, expressed in decimal degrees using floating point values, using the WGS84 datum.
`longitude`	–	Calculated physical geographic location, expressed in decimal degrees using floating point values, using the WGS84 datum.
`hpe`	–	Horizontal Positioning Error. Estimated horizontal error of the location, in meters, with 68% confidence. **NOTE** The default 68% confidence is supported in positioning service v2.28 and later. For 95% confidence, set `hpe-confidence` to 95 in the location request.
`altitude`	–	(Optional) Altitude above WGS84, in meters.
`vpe`	–	(Optional) Vertical Positioning Error. Estimated vertical error of the altitude, in meters, with 68% confidence.
`speed`	–	(Optional) Speed in m/s.
`bearing`	–	(Optional) Bearing, in degrees, from North in counterclockwise (+90 ° is West).
`age`	–	(Optional) Relative age of the measurement, in milliseconds or timestamp.
`speed-uncertainty`	–	(Optional) Uncertainty of speed, in m/s with 68% confidence.
`bearing-uncertainty`	–	(Optional) Uncertainty of bearing, in degrees with 68% confidence.
`constellation`	–	List of constellation types that contributed to the fix result.

Parameter		Values	Description
	`name`	–	(Optional) Name of the constellation of GNSS satellites. The possible examples are GPS, GLONASS, Galileo, BeiDou, QZSS, SBAS, and UNKNOWN.
	`nsat`	–	(Optional) Number of GNSS satellites, an integer value, used in a constellation to compute location.

3.3 Scanned BLE beacon group

The scanned Bluetooth low energy (BLE) beacon group allows the input of observed BLE beacons. Multiple BLE beacons and multiple types of BLE beacons are allowed as input.

3.3.1 IBeacon BLE

Provides the iBeacon BLE beacon observed during a scan.

Syntax

```
<ibeacon-ble>
  <uuid>123e4567-e89b-12d3-A456-426655440000</uuid>
  <major>1</major>
  <minor>2</minor>
  <company-id>666</company-id>      <!-- Optional -->
  <mac>000C41A2DF52</mac>        <!-- Optional -->
  <rssi>-95</rssi>               <!-- Optional -->
  <age>1000</age>              <!-- Optional -->
  <tx-power>-50</tx-power>          <!-- Optional -->
</ibeacon-ble>
```

Parameters

Parameter	Values	Description
`uuid`	–	Universally Unique Identifier. Comprises 32 hexadecimal digits, split into 5 groups, separated by hyphens; 16 bytes.
`major`	Range: 0 to 65535	Additional identifying number assigned to iBeacon; unsigned integer.
`minor`	Range: 0 to 65535	Additional identifying number assigned to iBeacon; unsigned integer.
`company-id`	–	(Optional) Identifier provided with iBeacon prefix; integer.
`mac`	–	(Optional) MAC address of the device.
`rssi`	–	(Optional) Signal strength received from the beacon, in dBm. **NOTE** If this value is provided, `tx-power` must also be provided for higher accuracy.

Parameter	Values	Description
`age`	–	(Optional) Relative age of the scan, in milliseconds or timestamp.
`tx-power`	Range: -128 to 127	(Optional) Transmission power of the beacon; `rx_power` at 1 m from BLE beacon (1 byte); signed integer.

3.3.2 Eddystone BLE

Provides the Eddystone BLE beacon observed during a scan.

NOTE Only Eddystone BLE beacons with unique identifiers are supported.

Syntax

```
<eddystone-uid-ble>
  <namespace-id>123e4567e89b12d3A456</namespace-id>
  <instance-id>426655440000</instance-id>
  <mac>000C41A2DF52</mac>                <!-- Optional -->
  <rssi>-95</rssi>                       <!-- Optional -->
  <age>1000</age>                        <!-- Optional -->
  <ranging-data>-50</ranging-data>             <!-- Optional -->
</eddystone-uid-ble>
```

Parameters

Parameter	Values	Description
`namespace-id`	–	10-byte hexadecimal identifier; string.
`instance-id`	–	6-byte hexadecimal identifier; string.
`mac`	–	(Optional) MAC address of the device.
`rssi`	–	(Optional) Signal strength received from the beacon, in dBm. **NOTE** If this value is provided, `tx-power` must also be provided for higher accuracy.
`age`	–	(Optional) Relative age of the scan, in milliseconds or timestamp.
`ranging-data`	–	(Optional) Configured maximum transmission power, in dBm, emitted by the beacon at 0 m (1 byte).

3.3.3 AltBeacon BLE

Provides the AltBeacon BLE beacon observed during a scan.

Syntax

```
<altbeacon-ble>
  <uuid>123e4567-e89b-12d3-A456-426655440000</uuid>
  <major>1</major>
  <minor>2</minor>
```

```
  <company-id>666</company-id>              <!-- Optional -->
  <mac>000C41A2DF52</mac>                   <!-- Optional -->
  <rssi>-95</rssi>                        <!-- Optional -->
  <age>1000</age>                         <!-- Optional -->
  <ref-rssi>-50</ref-rrsi>                  <!-- Optional -->
<altbeacon-ble>
```

Parameters

Parameter	Values	Description
`uuid`	–	Universally Unique Identifier. Comprises 32 hexadecimal digits, split into 5 groups, separated by hyphens; 16 bytes.
`major`	Range: 0 to 65535	Additional identifying number assigned to AltBeacon; unsigned integer.
`minor`	Range: 0 to 65535	Additional identifying number assigned to AltBeacon; unsigned integer.
`company-id`	–	(Optional) Identifier provided with AltBeacon prefix; integer.
`mac`	–	(Optional) MAC address of the device.
`rssi`	–	(Optional) Signal strength received from the beacon, in dBm. **NOTE** If this value is provided, `ref-rssi` must also be provided for higher accuracy.
`age`	–	(Optional) Relative age of the scan, in milliseconds or timestamp.
`ref-rssi`	Range: -127 to 0	(Optional) Average signal strength received at 1 m from the advertiser; signed 1-byte value.

4 Positioning service XML API response

The positioning service XML API response includes latitude, longitude, and HPE (accuracy) that are determined from the positioning service XML API request inputs.

NOTE Qualcomm Technologies, Inc. (QTI) uses multiple methods and sources for positioning IPs. Hence, HPE may not be returned in all responses.

Syntax

Response messages are defined by the following syntax:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<LocationRS version="2.31" xmlns="http://skyhookwireless.com/wps/2005">
    <location age="0" ncell="3" lap="-P0Y2M8D" nap="3">
        <latitude>##.##</latitude>
        <longitude>##.##</longitude>
        <hpe>##</hpe>
        <street-address>    <!-- Optional --> <!-- See below for details -->
        </street-address>
        <timezone>          <!-- Optional --> <!-- See below for details -->
        </timezone>
    </location>
</LocationRS>
```

Parameters

Parameter	Description
`version`	The version of the API being used.
`rqtime`	The time at which the XML API request was sent. This value is a reflection of the value provided in the request.
`age`	Relative age of the location, in milliseconds or timestamp.
`ncell`	Number of cell towers sent in the request known to the server.
`nlac`	Number of Location Area Codes (LACs) sent in the request known to the server.
`lap`	Age of the youngest access point used to determine the location.
`nap`	Number of access points sent in the request known to the server.
`nsat`	Number of satellites used to compute the GPS location supplied by the client, if such information was used to determine location.
`nble`	Number of BLE beacons sent in the request known to the server. This parameter is available in positioning service v2.31 and later versions.

Parameter		Description
`location-source`		Source of the location. Possible values are: ■ `unknown` ■ `cell` ■ `wifi` ■ `gnss` ■ `ble` ■ `rfpm` ■ `hybrid` (scans of multiple beacon types used) This parameter is available in positioning service v2.29 and later versions.
`latitude`		Calculated physical geographic location, expressed in decimal degrees using floating point values, using the WGS84 datum.
`longitue`		Calculated physical geographic location, expressed in decimal degrees using floating point values, using the WGS84 datum.
`hpe`		Estimates HPE of the location, in meters, when `hpeConfidence` is set to 68% (default) or 95% in the API request. **NOTE** The default 68% confidence is supported in positioning service v2.28 and later. For 95% confidence, set `hpe-confidence` to 95 in the location request.
`altitude`		Calculated z-axis position of the location, in meters. This value is returned only if barometric readings are provided in the request.
	`ref-type`	The reference at which the altitude value is with respect to WGS84, Above Ground Level (AGL), Mean Sea Level (MSL).
	`vpe`	Vertical Positioning Error (VPE) of the altitude value. Estimated vertical error of the altitude, in meters, with 68% confidence.
`street-address`		Calculated street position of the location. A value is returned only if `street-address-lookup-type` is set to either `full` or `limited` in the API request.
	`distanceToPoint`	Distance from the returned location to the address.
	`street-number`	Street number of the address.
	`address-line`	Contains the street name, and other address lines such as apartment number, if applicable.
	`neighborhood`	Neighborhood of the address.
	`metro1`	City of the location.
	`metro2`	City/township of the location.
	`postal-code`	Postal code associated with the address.
	`county`	County in which the address is located.
	`province`	Province in which the address is located.
	`region`	Region in which the address is located.
	`state`	State in which the address is located. Code is the relevant code of the state.
	`country`	Country code in which the address is located. Code is the relevant code of the country.

Parameter		Description
`timezone`		Calculated timezone of the location. A value is returned only if `timezone-lookup` is set to `true` in the API request.
	`name`	Name of the time zone.
	`tz-offset`	UTC offset of the time zone.
	`dst-offset`	Daylight savings offset of the time zone.

Sample response

```
<location rqtime="2496732469823" age="350" ncell="3" lap="-P0Y2M8D"
nlac="2" nap="3" nsat="3" nble="2" location-source="hybrid">
    <latitude>35.8723467378</latitude>
    <longitude>38.3628442</longitude>
    <hpe>12</hpe>
    <street-address>               <!-- Optional -->
        <address-line>George Street</address-line>
        <metro1>Sydney</metro1>
        <metro2>Haymarket</metro2>
        <postal-code>2000</postal-code>
        <state code="NSW">New South Wales</state>
        <country code="AU">Australia</country>
    </street-address>
    <timezone>
        <name>America/Mexico_City</name>
        <tz-offset>-PT6H</tz-offset>
        <dst-offset>PT1H</dst-offset>
    </timezone>
</location>
```

Warnings and errors

A partially successful request to the server returns one or more warning messages with the following syntax:

```
<LocationRS>WARNING</LocationRS>
```

An unsuccessful request to the server returns an error message with the following syntax:

```
<LocationRS version="2.28" xmlns="http://skyhookwireless.com/wps/2005">
    <error>Unable to determine location</error>
</LocationRS>
```

Where:

- `string-with-code` is the textual description of the warning or error.
- `code` is the numeric code of the warning or error.

5 Test sample positioning service XML API request

To issue a positioning service XML API request or to test using Python or cURL commands, perform the following steps:

1. Create a file named `location_rq.xml` with the following sample XML request body:

```
<LocationRQ xmlns="http://skyhookwireless.com/wps/2005"
            version="2.31"
            street-address-lookup="none">
<authentication version="2.2">
<key key="YOUR API KEY HERE"
     username="DEVICE SERIAL NUM, MAC ADDRESS, OR OTHER UNIQUE ID HERE"/>
</authentication>
   <access-point>
        <mac>E2557D913550</mac>
        <ssid>Skyhook-1400</ssid>
        <signal-strength>-41</signal-strength>
        <age>1000</age>
        <freq>2437</freq>
    </access-point>
    <access-point>
        <mac>E2557D60E470</mac>
        <ssid>Skyhook-1400</ssid>
        <signal-strength>-43</signal-strength>
        <age>1000</age>
        <freq>5520</freq>
    </access-point>
    <access-point>
        <mac>E2556D63C520</mac>
        <ssid>Skyhook-1400</ssid>
        <signal-strength>-70</signal-strength>
        <age>1000</age>
        <freq>2462</freq>
    </access-point>
    <lte-tower>
        <mcc>311</mcc>
        <mnc>480</mnc>
        <tac>25828</tac>
```

```
            <eucid>25919502</eucid>
            <pci>210</pci>
            <rssi>-50</rssi>
            <timing-advance>0</timing-advance>
            <age>0</age>
            <earfcn>1000</earfcn>
            <serving>true</serving>
        </lte-tower>
        <ibeacon-ble>
            <uuid>f7826da6-4fa2-4e98-8024-bc5b71e0893e</uuid>
            <major>58966</major>
            <minor>39642</minor>
            <company-id>1400</company-id>
            <mac>F7293682CB2B</mac>
            <rssi>-89</rssi>
            <age>280</age>
            <tx-power>-77</tx-power>
        </ibeacon-ble>
        <eddystone-uid-ble>
            <namespace-id>f7826da6bc5b71e0893e</namespace-id>
            <instance-id>6a573133467a</instance-id>
            <mac>CF890377C033</mac>
            <rssi>-90</rssi>
            <age>1000</age>
            <ranging-data>-36</ranging-data>
        </eddystone-uid-ble>
    </LocationRQ>
```

2. To test using Python, submit the file using the following Python script:

   ```
   python WPS_API.py   C:\path\to\file\location_rq.xml
   ```

   Ensure that the `requests` library is available for the script to run properly.

   ```
   #WPS_API.py

   import sys
   import os.path
   import requests

   # Need file name, index of ip & newline
   if len (sys.argv) < 2:
           print "USAGE: {} <xml filename>".format(sys.argv[0])
           exit(-1)
   in_filename = sys.argv[1]

   # Open input and output file streams
   in_file = open(in_filename, "r")

   # Begin processing IP addresses
   ```

```
print "Script started"
print "Analyzing Log File"

# Read in xml file
xml_string = in_file.read()

# HTML headers
headers = {'Content-Type': 'text/xml'}


# Make the request to the Precision location API
r = requests.post('https://global.skyhookwireless.com/wps2/location',
data=xml_string,
headers=headers)print r.text

print "Script finished"

# Close input and output file streams
in_file.close()
```

3. To test using cURL, submit the file using a cURL command-line tool with the following syntax (all in one line):

```
curl -H "Content-Type: text/xml" -H "X-Forwarded-For: 127.0.0.1" -d
@location_rq.xml https://global.skyhookwireless.com/wps2/location
```

NOTE Ensure that the command is formatted properly without any whitespaces when using cURL.

6 Guidelines to evaluate positioning service XML API

Before testing the positioning service XML API on a large scale, it is necessary to evaluate the accuracy and yield of Wi-Fi and cellular ID-based services correctly.

6.1 Guidelines to perform field test and collect data

The best approach for testing the positioning service XML API is to conduct testing in the field where real-world errors can be determined. Qualcomm has the tools and applications required to help with field testing, data collection, and analysis.

Consider the following guidelines and recommendations to achieve consistent and reliable test results:

- Test in various morphologies such as dense urban, suburban, deep indoor, and indoor.
  - A mix of 70% indoors and 30% outdoors is recommended to account for devices being indoors most of the day.
- A statistically significant sample size is required to get an accurate picture of the location services.
  - At a minimum, take several thousand samples in a given city. If this is not achievable due to limitations in resources or individual city access, contact the Qualcomm Aware™ Positioning Service team to determine a fair sample size and location criteria for the specific test area.
- When choosing test points, ensure that an accurate ground truth location can be determined and recorded.
  - Use street corners or other recognizable landmarks for testing. Online satellite and vector map services can help to determine ground truth.
- After selecting the test point, ensure that the device is not in motion and record the following information:
  - Time
  - Real location (ground truth), where the test sample is taken via a mapping service
  - Result of location system
- After recording all the information of the test location, compare each location result to the ground truth to determine individual location errors.
  - Compile these individual location errors into a Cumulative Distribution Function (CDF). See Guidelines to analyze test results for more information.

6.2 Guidelines to use GNSS/GPS for ground truth

With recent improvements in GNSS and receiver technologies, GNSS/GPS is often used to determine accurate ground truth. However, using GNSS/GPS for ground truth is not advised in many cases as it can be inaccurate in urban areas due to the urban-canyon effect of multipath. In such cases, it is possible that the GNSS receiver suggests that its accuracy is good, but in reality it is not.

Performing field tests particularly on a large scale may not always be possible due to constraints in resources, time, and cost. However, high-quality GNSS reference points may mostly be available in areas where network location is least valuable such as outdoors in areas with clear view of the sky.

When determining the quality of a location system using GNSS/GPS, consider the following guidelines and recommendations to achieve consistent and reliable test results. These results are used for replaying RF signal environment scans and comparing them to GNSS as a proxy for accurate ground truth.

- Select a bounding box for the test area with enough device traffic and assumed access point/cell coverage to remove signal scan and data processing variability.
  - This bounding box can be in metropolitan areas or suburban environments where network-based locations are most valuable to a system but also have the potential for accurate GNSS reference locations.
- Ensure that the GNSS samples used indicate that the location result is highly accurate.
  - Use GNSS samples where recommended GNSS uncertainty is $\leq$ 10 m and the number of satellites reporting is $\geq$ 8.
  - Remove GNSS samples that indicate that the device is likely in motion. This includes GNSS attributes such as speed < 0.5 m/s.
- Ensure that there is minimal delta time between the GNSS fix used and the Wi-Fi/network information replayed for that sample.
  - For example, the recommended delta time between the GNSS fix and the corresponding Wi-Fi scan is $\leq$ 1.
- Ensure that the data use for replays is as recent as possible.
  - In general, Wi-Fi access points are moved and reused, and cell identifiers can be rotated or changed by operators. These updated locations are reflected in the most recent database of location system providers.
  - Using the most recent possible data reduces the risk that access points and cell IDs may have either moved or changed during the time between data collection and the replayed requests.

6.3 Guidelines to analyze test results

Comparing results from different location systems can be difficult due to varying algorithms and trade-offs between yield and accuracy. Here, yield is the percentage of time a location is returned and accuracy is the distance to ground truth.

A few common industry standard methods for improving accuracy metrics over yield are as follows:

- Do not include an error value for individual location requests that produced no or poor results.
- Do not include an error value for individual location requests for single access point locations or low number of access points.

Qualcomm implements algorithms and fallback logic to maximize yield and provides the best location solution suitable for all circumstances. Compared to returning a failed location in most cases, this solution provides the best possible location of any sort. OEMs can use the best available location result depending on the use case, and if necessary, filter out locations that are above a certain error estimation value.

Though it is difficult to compare different systems, it is a necessary step to choose a technology. Consider the following guidelines and recommendations to compare systems as equally possible:

- Select a bounding box for the test area with enough device traffic and assumed access point/cell coverage to remove signal scan and data processing variability.
  - This bounding box can be in metropolitan areas or suburban environments where network-based locations are most valuable to a system but also have the potential for accurate GNSS reference locations.
- Ensure that the request or test sample size of a given region is large enough to evaluate the systems appropriately.
  - Though it depends on the available data, time, or bandwidth of the tester, it is recommended to take several thousand samples for each region evaluated.
  - Sample counts should accompany test results to understand the scale at which the solutions were tested in each region.
- Ensure to evaluate both accuracy and yield in a given region.
- To analyze accuracy, calculate the distance to ground truth (error) in each individual location response.
  - Use these values to calculate and plot a CDF in increments of 10% across the entire set for each test region or in aggregate.
  - For example, an accuracy value of X meters at the 50th percentile indicates 50% of locations in the set are either equal to or less than error X.
  - Include the 67% as this represents 1-sigma or 1 standard deviation from the statistical mean.
- Standardize an error value for each failed location result.
  - It is important to not skew accuracy data in higher percentiles for locations that can only be returned for a given system.
  - Set a large distance for this error value, for example, 100,000 m or greater.

A References

Table A-1 Acronyms and terms

Acronym or term	Definition
AES	Advanced encryption standard
BLE	Bluetooth low energy
CDF	Cumulative distribution function
CDMA	Code division multiple access
GPS	Global positioning system
GNSS	Global navigation satellite system
GSM	Global system for mobile
HPE	Horizontal position error
LTE	Long-term evolution
NB-IoT	Narrowband Internet of things
NR	New radio
RTT	Round-trip time
SSID	Service set identifier
UMTS	Universal mobile telecommunications service
WCDMA	Wideband code division multiple access

LEGAL INFORMATION

Your access to and use of this material, along with any documents, software, specifications, reference board files, drawings, diagnostics and other information contained herein (collectively this "Material"), is subject to your (including the corporation or other legal entity you represent, collectively "You" or "Your") acceptance of the terms and conditions ("Terms of Use") set forth below. If You do not agree to these Terms of Use, you may not use this Material and shall immediately destroy any copy thereof.

1) Legal Notice.

This Material is being made available to You solely for Your internal use with those products and service offerings of Qualcomm Technologies, Inc. ("**Qualcomm Technologies**"), its affiliates and/or licensors described in this Material, and shall not be used for any other purposes. If this Material is marked as "**Qualcomm Internal Use Only**", no license is granted to You herein, and You must immediately (a) destroy or return this Material to Qualcomm Technologies, and (b) report Your receipt of this Material to qualcomm.support@qti.qualcomm.com. This Material may not be altered, edited, or modified in any way without Qualcomm Technologies' prior written approval, nor may it be used for any machine learning or artificial intelligence development purpose which results, whether directly or indirectly, in the creation or development of an automated device, program, tool, algorithm, process, methodology, product and/or other output. Unauthorized use or disclosure of this Material or the information contained herein is strictly prohibited, and You agree to indemnify Qualcomm Technologies, its affiliates and licensors for any damages or losses suffered by Qualcomm Technologies, its affiliates and/or licensors for any such unauthorized uses or disclosures of this Material, in whole or part.

Qualcomm Technologies, its affiliates and/or licensors retain all rights and ownership in and to this Material. No license to any trademark, patent, copyright, mask work protection right or any other intellectual property right is either granted or implied by this Material or any information disclosed herein, including, but not limited to, any license to make, use, import or sell any product, service or technology offering embodying any of the information in this Material.

THIS MATERIAL IS BEING PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESSED, IMPLIED, STATUTORY OR OTHERWISE. TO THE MAXIMUM EXTENT PERMITTED BY LAW, QUALCOMM TECHNOLOGIES, ITS AFFILIATES AND/OR LICENSORS SPECIFICALLY DISCLAIM ALL WARRANTIES OF TITLE, MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR A PARTICULAR PURPOSE, SATISFACTORY QUALITY, COMPLETENESS OR ACCURACY, AND ALL WARRANTIES ARISING OUT OF TRADE USAGE OR OUT OF A COURSE OF DEALING OR COURSE OF PERFORMANCE. MOREOVER, NEITHER QUALCOMM TECHNOLOGIES, NOR ANY OF ITS AFFILIATES AND/OR LICENSORS, SHALL BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY EXPENSES, LOSSES, USE, OR ACTIONS HOWSOEVER INCURRED OR UNDERTAKEN BY YOU IN RELIANCE ON THIS MATERIAL.

Certain product kits, tools and other items referenced in this Material may require You to accept additional terms and conditions before accessing or using those items.

Technical data specified in this Material may be subject to U.S. and other applicable export control laws. Transmission contrary to U.S. and any other applicable law is strictly prohibited.

Nothing in this Material is an offer to sell any of the components or devices referenced herein.

This Material is subject to change without further notification.

In the event of a conflict between these Terms of Use and the *Website Terms of Use* on www.qualcomm.com, the *Qualcomm Privacy Policy* referenced on www.qualcomm.com, or other legal statements or notices found on prior pages of the Material, these Terms of Use will control. In the event of a conflict between these Terms of Use and any other agreement (written or click-through, including, without limitation any non-disclosure agreement) executed by You and Qualcomm Technologies or a Qualcomm Technologies affiliate and/or licensor with respect to Your access to and use of this Material, the other agreement will control.

These Terms of Use shall be governed by and construed and enforced in accordance with the laws of the State of California, excluding the U.N. Convention on International Sale of Goods, without regard to conflict of laws principles. Any dispute, claim or controversy arising out of or relating to these Terms of Use, or the breach or validity hereof, shall be adjudicated only by a court of competent jurisdiction in the county of San Diego, State of California, and You hereby consent to the personal jurisdiction of such courts for that purpose.

2) Trademark and Product Attribution Statements.

Qualcomm is a trademark or registered trademark of Qualcomm Incorporated. Arm is a registered trademark of Arm Limited (or its subsidiaries) in the U.S. and/or elsewhere. The Bluetooth® word mark is a registered trademark owned by Bluetooth SIG, Inc. Other product and brand names referenced in this Material may be trademarks or registered trademarks of their respective owners.

Snapdragon and Qualcomm branded products referenced in this Material are products of Qualcomm Technologies, Inc. and/or its subsidiaries. Qualcomm patented technologies are licensed by Qualcomm Incorporated.