Open API Project

Aepona v1.0 Sandbox Data Service REST

Document Version 1.1

Document Revision History

Rev #	Date	Description
1.0	Nov 1 2013	Initial version of API_Aepona1-0_SandboxDataService-REST_USCC based on Aepona ASE 1.1 SDS v1.0 API Guide doc v1.2.
1.1	Nov 14 2013	Updated details to match production platform.

Copyright

Table of Contents

1 Sandbox Data Service REST Overview

The Sandbox Data Service allows developers to create groups, as well as subscribers and their attributes. This data is used to configure a Sandbox services' requests or responses to the developer's application. This allows the developer to execute a number test scenarios with their application.

A server side certificate is required plus HTTP Basic Authentication.

For more information, refer to the "Developer Access' section in the 'OneAPI v2.0 Common Information Guide'.

> **!** Data types indicated as XSD in parameter tables refer to the standard XSD type. Relevant information may be obtained from *http://www.w3.org/2001/XMLSchema*.

2 Methods

Sandbox Data Service may be accessed via the REST API (described in this document). The following methods are available:

- View Groups
- Add Group
- Create & View Subscribers:
  - Create Subscriber for Specific Group
  - Populate Subscriber
  - View Subscriber
  - View Subscriber With Filter
- Delete Subscriber:
  - Delete Subscriber Object
  - Delete a Subscriber Field/Attribute

POST, GET and DELETE HTTP methods are used in Sandbox Data Service.

2.1 URIs

The URIs for the resources are as follows:

- View groups

  https://developerportal.uscellular.com/services/sandboxDataService

  HTTP METHOD: GET

- Add Group

  https://developerportal.uscellular.com/services/sandboxDataService/{group_name}

  HTTP METHOD: POST

- Create subscriber

  https://developerportal.uscellular.com/services/sandboxDataService/{group_name}/{subscriber}

  HTTP METHOD: POST

- View subscriber(s) within a group

  https://developerportal.uscellular.com/services/sandboxDataService/{group_name}

  HTTP METHOD: GET

- Delete subscribers

  https://developerportal.uscellular.com/services/sandboxDataService/{group_name}/{subscriber}

  HTTP METHOD: DELETE

The variables used in request URLs are described below:

Name	Description
{group_name}	The sandbox group name. See 'Available Sandboxes' on page 27 for the possible group_name values.

{subscriber}	Address (CTN) of U.S. Cellular subscriber, in the format comprising: 1) 'tel:' protocol identifier, 2) country code of one preceded by '+', 3) ten digit CTN preceded by 1, for example, tel:+15087300001

! Throughout this document, the examples may be shown WITHOUT URL encoding for readability purposes.

! The *application/xml* content type is supported in the responses.

3 View Groups

This lets you display the list of groups created.

3.1.1 Request

```
GET https://developerportal.uscellular.com/services/SandboxDataService/
HTTP/1.1
Authorization: Basic VExYX1JBWV8xMDE4OlBBU1NXT1JE
User-Agent: Jakarta Commons-HttpClient/3.1
Host: developerportal.uscellular.com
```

3.1.2 Request Parameters

N/A

3.1.3 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Date: Tue, 19 Oct 2010 08:28:38 GMT
Content-Length: 169
Server: Jetty(6.1.x)
<?xml version="1.0" encoding="UTF-8" standalone="yes”?>
<groups>
        <groupName>MMS_MO_Sandbox</groupName>
        <groupName>SMS_MO_Sandbox</groupName>
        <groupName>Terminal_Location_Sandbox</groupName>
        <groupName>MMS_MT_Sandbox</groupName>
        <groupName>SMS_MT_Sandbox</groupName>
```



```
</groups>
```

3.1.4 Response Parameters

Table 1: View Groups Response Parameters (groups Type)

Parameter	Type	Description	Optional
group_name	xsd:string	The name of the group within the sandbox data service. See 'Available Sandboxes' on page 27 for the possible group_name values.	Yes

4 Add Group

This method allows you to create a new group.

4.1.1 Request

```
POST
https://developerportal.uscellular.com/services/SandboxDataService/Terminal_Location
_Sandbox

HTTP/1.1

Content-Type: application/x-www-form-urlencoded

User-Agent: http4e/5.0.2

Authorization: Basic c2FuZGJveHBhcnRuZXI6cGFzc3dvcmQ=

Content-Length: 0

Host: developerportal.uscellular.com
```

4.1.2 Request Parameters

Table 2: Add Group Request Parameters

Parameter	Type	Description	Optional
group_name	xsd:string	The name of the group. In the sample above, this group is called: Terminal_Location_Sandbox. See 'Available Sandboxes' on page 27 for the possible group_name values.	No

4.1.3 Response

```
HTTP/1.1 201 Created

Content-Type: application/xml

Date: Thu, 15 Mar 2012 05:31:11 GMT
```



```
Content-Length: 0

Server: Jetty(6.1.x)
```

4.1.4 Response Parameters

N/A

5 Create & View Subscribers

5.1 Create Subscriber for Specific Group

This method allows you to create a new subscriber and add them to a specific group.

5.1.1 Request

```
POST
https://developerportal.uscellular.com/services/sandboxDataService/Terminal_Location
_Sandbox/tel:+12345678900

HTTP/1.1

Content-Type: application/x-www-form-urlencoded

Authorization: Basic VExYX1JBWV8xMDE4OlBBU1NXT1JE

User-Agent: Jakarta Commons-HttpClient/3.1

Content-Length: 0

Host: developerportal.uscellular.com
```

5.1.2 Request Parameters

Table 3: Create Subscriber Request Parameters

Parameter	Type	Description	Optional
group_name	xsd:string	The name of the group to which you want to add the subscriber. In the sample above, this group is called: Terminal_Location_Sandbox. See 'Available Sandboxes' on page 27 for the possible group_name values.	No
subscriber	xsd:string	The subscriber being created. In the sample above, this subscriber is called: tel: +12345678900.	No

5.1.3 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Date: Mon, 18 Oct 2010 15:40:35 GMT
Content-Length: 0
Server: Jetty(6.1.x)
```

5.1.4 Response Parameters

N/A

5.2 Populate Subscriber

This method lets you populate a subscriber object. Subscriber fields differ depending on the service. See 'Available Sandboxes' on page 27 for more information.

5.2.1 Request

```
POST
https://developerportal.uscellular.com/services/sandboxDataService/Terminal_Location
_Sandbox/tel:+12345678900?accuracy=10&altitude=10&longitude=12&latitude=12
HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Authorization: Basic VExYX1JBWV8xMDE4OlBBU1NXT1JE
User-Agent: Jakarta Commons-HttpClient/3.1
Content-Length: 0
Host: developerportal.uscellular.com
```


5.2.2 Request Parameters

Table 4: Populate Subscriber Request Parameters

Parameter	Type	Description	Optional
group_name	xsd:string	The name of the group to which you want to add the subscriber. In the sample above, this group is called: Terminal_Location_Sandbox. See 'Available Sandboxes' on page 27 for the possible group_name values.	No
subscriber	xsd:string	The subscriber being created. In the sample above, this subscriber is called: tel: +12345678900.	No

! Parameters depend on the service being tested. There is no validation to check that the appropriate parameters are included for the service. The sample above is for the Terminal Location service.

5.2.3 Response

```
HTTP/1.1 201 Created
Content-Type: application/xml
Date: Tue, 19 Oct 2010 08:28:38 GMT
Content-Length: 169
Server: Jetty(6.1.x)
```

5.2.4 Response Parameters

N/A

5.3 View Subscribers

This method allows you to display the details of a subscriber from a specific group or all subscribers in a group.

5.3.1 Request

```
GET
https://developerportal.uscellular.com/services/sandboxDataService/Terminal_Location
_Sandbox/tel:+12345678900

HTTP/1.1

Authorization: Basic VExYX1JBWV8xMDE4OlBBU1NXT1JE

User-Agent: Jakarta Commons-HttpClient/3.1

Host: developerportal.uscellular.com
```

5.3.2 Request Parameters

Table 5: View Subscriber Request Parameters

Parameter	Type	Description	Optional
group_name	xsd:string	The name of the group with which the subscriber is associated. In the sample above, this group is called: Terminal_Location_Sandbox. See 'Available Sandboxes' on page 27 for the possible group_name values.	No
subscriber	xsd:string	The subscriber to view. In the sample code above, this value is 'tel:+12345678900'. Leaving this blank, i.e., not adding / {subscriber}, will display all of the subscribers for the queried group.	Yes

5.3.3 Response

```
HTTP/1.1 200 OK

Content-Type: application/xml

Date: Tue, 19 Oct 2010 08:28:38 GMT

Content-Length: 169
```



```
Server: Jetty(6.1.x)
<?xml version="1.0" encoding="UTF-8" standalone="yes”?>
<subscriber address="tel:+12345678900">
  <attribute value="10" name="accuracy"/>
  <attribute value="12" name="latitude"/>
  <attribute value="10" name="altitude"/>
  <attribute value="12" name="longitude"/>
</subscriber>
```

5.3.4 Response Parameters

Table 6: View Subscriber Response Parameters

Parameter	Type	Description	Optional
subscriber	xsd:string	The subscriber associated with the group.	Yes

! Response parameters depend on the service being tested. The sample above is for the Terminal Location service.

5.4 View Subscriber With Filter

This method allows you to display specific subscriber fields based on a given filter.

5.4.1 Request

```
GET
http://developerportal.uscellular.com/services/sandboxDataService/Terminal_Location_
Sandbox/tel:+12345678900?filter=accuracy
HTTP/1.1
Accept: application/xml
Authorization: Basic VExYX1JBWV8xMDE4OlBBU1NXT1JE
User-Agent: Jakarta Commons-HttpClient/3.1
```



```
Host: developerportal.uscellular.com
```

5.4.2 Request Parameters

Table 7: View Subscriber With Filter Request Parameters

Parameter	Type	Description	Optional
group_name	xsd:string	The The name of the group with which the subscriber is associated. In the sample above, this group is called: Terminal_Location_Sandbox. See 'Available Sandboxes' on page 27 for the possible group_name values.	No
subscriber	xsd:string	The subscriber to view. In the sample code above, this value is 'tel:+12345678900'.	Yes
filter	xsd:string	Attribute of the subscriber you want to view. See 'Available Sandboxes' on page 27 for more information.	Yes

5.4.3 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Date: Tue, 19 Oct 2010 08:28:38 GMT
Content-Length: 169
Server: Jetty(6.1.x)
<?xml version="1.0" encoding="UTF-8" standalone="ye?>
<subscriber address="tel:+12345678900">
  <attribute value="10" name="accuracy"/>
</subscriber>
```


5.4.4 Response Parameters

Table 8: View Subscriber With Filter Response Parameters

Parameter	Type	Description	Optional
subscriber	xsd:string	The subscriber being queried.	Yes

6 Delete Subscriber

6.1 Delete Subscriber Object

This method removes the link between a subscriber and a group, effectively removing the subscriber object from the group.

6.1.1 Request

```
DELETE
https://developerportal.uscellular.com/services/sandboxDataService/Terminal_Location
_Sandbox/tel:+12345678900

HTTP/1.1

Content-Type: application/x-www-form-urlencoded

Authorization: Basic VExYX1JBWV8xMDE4OlBBU1NXT1JE

User-Agent: Jakarta Commons-HttpClient/3.1

Content-Length: 0

Host: developerportal.uscellular.com
```

! The **address** in the URL must be URL-escaped.

6.1.2 Request Parameters

Table 9: Delete Subscriber Request Parameters

Parameter	Type	Description	Optional
group_name	xsd:string	The name of the group with which the subscriber is associated. In the sample above, this group is called: Terminal_Location_Sandbox. See 'Available Sandboxes' on page 27 for the possible group_name values.	No

subscriber	xsd:string	The subscriber from whom you want to remove the link from the specified group. In the sample code above, this value is 'tel:+12345678900'.	No

6.1.3 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Date: Mon, 18 Oct 2010 15:40:35 GMT
Content-Length: 0
Server: Jetty(6.1.x)
```

6.1.4 Response Parameters

N/A

6.2 Delete a Subscriber Field/Attribute

This method allows you to remove a subscriber field or attribute.

6.2.1 Request

```
DELETE
https://developerportal.uscellular.com/services/sandboxDataService/Terminal_Location_Sandbox/tel:+12345678900?filter=accuracy
HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Authorization: Basic VExYX1JBWV8xMDE4OlBBU1NXT1JE
User-Agent: Jakarta Commons-HttpClient/3.1
Content-Length: 0
```



```
Host: developerportal.uscellular.com
```

! The **address** in the URL must be URL-escaped.

6.2.2 Request Parameters

Table 10: Delete Subscriber Field Request Parameters

Parameter	Type	Description	Optional
group_name	xsd:string	The name of the group with which the subscriber is associated. In the sample above, this group is called: Terminal_Location_Sandbox. See 'Available Sandboxes' on page 27 for the possible group_name values.	No
subscriber	xsd:string	The subscriber from whom you want to delete the attribute. In the sample code above, this value is 'tel:+12345678900'.	No
filter	xsd:string	Subscriber attribute to be removed. See 'Available Sandboxes' on page 27 for more information.	Yes

6.2.3 Response

```
HTTP/1.1 200 OK
Content-Type: application/xml
Date: Tue, 19 Oct 2010 08:28:38 GMT
Content-Length: 169
Server: Jetty(6.1.x)
```

6.2.4 Response Parameters

N/A

7 Response Codes & Exceptions

7.1 Response Codes

HTTP response codes are used to indicate:

- **200** – Success!
- **201** – Return when request is successfully created. Response code in adding of group and subscribers
- **400** – Bad request; check the error message for details
- **401** – Authentication failure, check your authentication details
- **403** – Forbidden; please provide authentication credentials
- **404** – Not found: mistake in the host or path of the service URI
- **405** – Method not supported: for example you mistakenly used a HTTP GET instead of a POST
- **500** – The server encountered an unexpected condition. It could be incorrect authentication details or limited user permission
- **503** – Server busy and service unavailable. Please retry the request.

For more details on these, refer to http://www.ietf.org/rfc/rfc2616.txt.

7.2 Exceptions

```
HTTP/1.1 401 Unauthorized
Content-type: application/xml
WWW-Authenticate: Basic realm=
Date: Wed, 20 Oct 2010 09:33:53 GMT
Content-Length: 91
Server: Jetty(6.1.x)
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<error> 401 - Not Authorized</error>
```


This section lists the available error codes, the possible reasons why the exception may have occurred, and possible solutions.

7.2.1 Service Exceptions

The following service exceptions may be thrown:

SVC0001 - Service error occurred

A service-related error has occurred as a result of a client invocation on the service. This category can be used for implementation-specific errors. Contact the support team.

7.2.2 Policy Exceptions

A policy exception means that the request syntax is valid, however an operator policy has been broken.

POL0001 – Policy error occurred

The above exception may be thrown to indicate a fault relating to a policy associated with the service. This category can be used for implementation-specific errors, which are listed below:

Table 11: Policy Error Codes

Error	Explanation
POL-006: TPA exceeded its maximum allowed rate of transactions	The maximum rate of transactions is exceeded. Ensure that the rate of your requests is within the limits set up in your SLA, e.g. 10 TPS (Transactions Per Second).
POL-008: TPA is invalid	The Third Party Application authentication details are incorrect. Check your basic authentication username and password are correct and re-submit your request.
POL-016: Max Requests is enforced, and max requests has been exceeded	The maximum number of requests for this service is exceeded. Contact the support team.

POL-017: Operation is not allowed	The method/operation is not supported in your current SLA. Check your SLA and use a method that is supported.

8 Available Sandboxes

The following sandboxes (and corresponding data) are available for use:

- MMS MO Sandbox
- MMS MT Sandbox
- SMS_MT_Sandbox
- SMS_MO Sandbox
- Terminal Location Sandbox

The parameters and provisioning scenarios for each of these sandboxes are listed in the following sections.

8.1 group_name: MMS_MO_Sandbox

The MMS_MO_Sandbox controls MMS notifications received when the notifications are set up on the Portal, with a number matching the subscriber number configured in the sandbox.

> **!** For the Sandbox notification scenarios to work, the subscriber number provisioned in the Sandbox must match the MMS destination address (= the number) set in the notification.

Table 12: MMS_MO_Sandbox Parameters

Parameter	Type	Description	Optional
maxNotifications	xsd:int	The maximum number of notifications that can be sent.	Yes
notificationDelay	xsd:int	The delay (in seconds) before the notification is available. Defaults to 10 seconds.	Yes

8.1.1 Provisioning Scenarios

Table 13: MMS_MO_Sandbox Provisioning Scenarios

Scenario	Result

No subscriber number provisioned	The notification criteria is not removed and remains in place. Use this scenario to test callbacks, as outlined in MMS MO Test Scenarios below.
Subscriber number provisioned with no data	A default of 1 notification is available after a delay of 10 seconds. The notification criteria is removed after it is sent.
Subscriber number provisioned with data	Provisioned data is returned, i.e. the number of notifications provisioned will be sent with the content of messageText and a delay of the provisioned time between them, based on the criteria and number (= MMS destination address) in the notification configured on the Portal. The notification criteria is removed after the last notification is sent.
Subscriber number provisioned with partial data, i.e. no maxNotifications data	A default of 1 notification will be sent if no maxNotifications data is provided. The notification criteria is removed after it is sent.
Subscriber number provisioned with partial data, i.e. no notificationDelay data	A default delay of 10 seconds between each notification will be used. The notification criteria is removed after the last notification is sent.

MMS MO Test Scenarios

MMS Sandbox services comprise the MMS send, receive and notification services with Sandbox adapters which can be used to test the operation of each service separately or together, as set out below.

- To test callbacks in isolation, in other words, without having to make a call to the MMS Send Sandbox endpoint, create a subscriber in the MMS_MO group with a number, and start a notification from the Portal using the same number. As outlined above, the MMS Service will look for an MMS_MO_Sandbox entry for the number, find it, and send the notification as per the defined criteria.

! Notification calls invoked from the Portal to the Sandbox are deleted after 30 seconds of the final notification having been sent. If you refresh the Portal you will no longer see the notification.

8.2 group_name: MMS_MT_Sandbox

The MMS_MT_Sandbox group defines what is returned when the Send MMS service is queried for delivery status.

8.2.1 Parameters

Table 14: MMS_MT_Sandbox Parameters

Parameter	Type	Description	Optional
deliveryStatus	deliveryStatus	The delivery status to be returned. Enumeration for deliveryStatus is as follows: • DeliveredToNetwork • DeliveredToTerminal • DeliveryUncertain • DeliveryImpossible • MessageWaiting	Yes
deliveryStatusDelay	xsd:int	The delay (in seconds) after making the send MMS request before the deliveryStatus becomes available. Defaults to 10 seconds.	Yes

8.2.2 Provisioning Scenarios

Table 15: MMS_MT_Sandbox Provisioning Scenarios

Scenario	Result
No subscriber number provisioned	The default delivery status of DeliveredToTerminal will be returned, provided the delivery status query is made after deliveryStatusDelay time. If the query is made before the deliveryStatusDelay, the delivery status of DeliveredToNetwork will be returned.

Subscriber number provisioned with no data	If the delivery status query is made after 10 seconds (default deliveryStatusDelay), the default delivery status of DeliveredToTerminal will be returned. If the query is made before 10 seconds, the default delivery status of DeliveredToNetwork will be returned.
Subscriber number provisioned with data,	If the delivery status query is made after the deliveryStatusDelay time set, the appropriate delivery status will be returned. If the query is made before the delay time, the default delivery status of DeliveredToNetwork will be returned.
Subscriber number provisioned with partial data, i.e. no deliveryStatusDelay data	If the delivery status query is made after 10 seconds (default deliveryStatusDelay time), the appropriate delivery status will be returned. If the query is made before 10 seconds, the delivery status of DeliveredToNetwork will be returned.

8.3 group_name: SMS_MO_Sandbox

The SMS_MO_Sandbox controls SMS notifications received when the notifications are set up on the Portal, with a number matching the subscriber number configured in the sandbox.

> **!** For the Sandbox notification scenarios to work, the subscriber number provisioned in the Sandbox must match the SMS destination address (= the number) set in the notification.

Table 16: SMS_MO_Sandbox Parameters

Parameter	Type	Description	Optional
maxNotifications	xsd:int	The maximum number of notifications that can be sent.	Yes
notificationDelay	xsd:int	The delay (in seconds) before the notification is available. Defaults to 10 seconds.	Yes
messageText	xsd:string	The content of the MO message to be sent to the application.	Yes

8.3.1 Provisioning Scenarios

Table 17: SMS_MO_Sandbox Provisioning Scenarios

Scenario	Result
No subscriber number provisioned	The notification criteria is not removed and remains in place. Use this scenario to test callbacks, as outlined in SMS MO Test Scenarios below.
Subscriber number provisioned with no data	A default of 1 notification is available after a delay of 10 seconds. The notification criteria is removed after it is sent.
Subscriber number provisioned with data	Provisioned data is returned, i.e. the number of notifications provisioned will be sent with

	the content of messageText and a delay of the provisioned time between them, based on the criteria and number (= SMS destination address) in the notification configured on the Portal. The notification criteria is removed after the last notification is sent.
Subscriber number provisioned with partial data, i.e. no maxNotifications data	A default of 1 notification will be sent if no maxNotifications data is provided. The notification criteria is removed after it is sent.
Subscriber number provisioned with partial data, i.e. no notificationDelay data	A default delay of 10 seconds between each notification will be used. The notification criteria is removed after the last notification is sent.

SMS MO Test Scenarios

SMS Sandbox services comprise the SMS send, receive and notification services with Sandbox adapters which can be used to test the operation of each service separately or together, as set out below.

- To test callbacks in isolation, in other words, without having to make a call to the SMS Send Sandbox endpoint, create a subscriber in the SMS_MO group with a number, and start a notification from the Portal using the same number. As outlined above, the SMS Service will look for an SMS_MO_Sandbox entry for the number, find it, and send the notification as per the defined criteria.
- To test the SMS service end-to-end, do not create a subscriber in the SMS_MO group. Just create a notification from the Portal, and make a call to SMS Send Sandbox endpoint with the destination address set to the number configured for the notification. Your application should receive a notification that the SMS has been received, and you will also be able to query the SMS Receive Sandbox endpoint to retrieve the message.

! Notification calls invoked from the Portal to the Sandbox are deleted after 30 seconds of the final notification having been sent. If you refresh the Portal you will no longer see the notification.

8.4 group_name: SMS_MT_Sandbox

The SMS_MT_Sandbox group defines what is returned when the Send SMS service is queried for delivery status.

8.4.1 Parameters

Table 18: SMS_MT_Sandbox Parameters

Parameter	Type	Description	Optional
deliveryStatus	delivery Status	The delivery status to be returned. Enumeration for deliveryStatus is as follows: • DeliveredToNetwork • DeliveredToTerminal • DeliveryUncertain • DeliveryImpossible • MessageWaiting	Yes
deliveryStatusDelay	xsd:int	The delay (in seconds) after making the send SMS request before the deliveryStatus becomes available. Defaults to 10 seconds.	Yes

8.4.2 Provisioning Scenarios

Table 19: SMS_MT_Sandbox Provisioning Scenarios

Scenario	Result
No subscriber number provisioned	The default delivery status of DeliveredToTerminal will be returned, provided the delivery status query is made after deliveryStatusDelay time. If the query is made before the deliveryStatusDelay, the delivery status of DeliveredToNetwork will be returned.

Subscriber number provisioned with no data	If the delivery status query is made after 10 seconds (default deliveryStatusDelay), the default delivery status of DeliveredToTerminal will be returned. If the query is made before 10 seconds, the default delivery status of DeliveredToNetwork will be returned.
Subscriber number provisioned with data,	If the delivery status query is made after the deliveryStatusDelay time set, the appropriate delivery status will be returned. If the query is made before the delay time, the default delivery status of DeliveredToNetwork will be returned.
Subscriber number provisioned with partial data, i.e. no deliveryStatusDelay data	If the delivery status query is made after 10 seconds (default deliveryStatusDelay time), the appropriate delivery status will be returned. If the query is made before 10 seconds, the delivery status of DeliveredToNetwork will be returned.

8.5 group_name: Terminal_Location_Sandbox

8.5.1 Parameters

Table 20: Terminal Location Sandbox Parameters

Parameter	Type	Description	Optional
latitude	xsd:float	The latitude of the geographical coordinates.	No
longitude	xsd:float	The longitude of the geographical coordinates.	No
altitude	xsd:float	The altitude of the geographical coordinates. If not specified, the value will be left blank.	Yes
accuracy	xsd:int	The accuracy of the location (in meters).	No

8.5.2 Provisioning Scenarios

Table 21: Terminal Location Sandbox Provisioning Scenarios

Scenario	Result
No subscriber number provisioned	Default data is returned (latitude=54.6575; longitude=-6.2158; altitude=10.0; accuracy=30); but is configurable in container properties.
Subscriber number provisioned with no location data	*'No data provisioned for this subscriber'* error returned.
Subscriber number provisioned with location data	Provisioned data will be returned in the response.
Subscriber number provisioned with partial data, i.e. no altitude data	Provisioned data will be returned with the altitude parameter left blank.

