# Event API

This chapter describes the Event API.

# Using the Event API

In your IoT FND NB API client application, use this IoT FND server URL to access the Event API WSDL:

http://<*server_address*> /nbapi/event?wsdl

For example:

http://10.27.167.19/nbapi/event?wsdl

# Event API Method Calls

This section contains the following topics:

## searchEvents

This call searches for events based on device type, event name, event time, and event severity. This API returns only the events of the domain that the user belongs to.

**Prototype**

```
<even:searchEvents>
<query>deviceType:cgmesh eventName:up</query>
<count>4</count>
<offset>0</offset>
</even:searchEvents>
```

## Parameters

*Table 1: searchEvents Parameters*

| Parameter | Type | Description |
|---|---|---|
| query | string | Search query string. |
| count | integer | Number of results to retrieve. |
| offset | integer | Position of the first result. |

Use the parameters in Table 2: Query Parameters, on page 2 and the options listed in Table 3: eventName Query Options , on page 3 in the query.

*Table 2: Query Parameters*

| Parameter | Delimiters | Options | Description |
|---|---|---|---|
| deviceType | = | asr1000<br><br>cgmesh<br><br>cgnms<br><br>cgr1000<br><br>db<br><br>c800<br><br>ir800<br><br>ir500<br><br>ap800 | Device type. |
| eventName | = | Options are listed in Table 3: eventName Query Options , on page 3 . | User-defined name. |
| eventSeverity | = | CRITICAL<br><br>MAJOR<br><br>MINOR<br><br>INFO | Severity level. |
| eventTime | >|<|>=|<= | User defined. | UTC date and time in the format: yyyy-MM-dd HH:mm:ss:SSS. |

**Table 3: eventName Query Options**

| Option | Device | Description |
|---|---|---|
| aaaFailure | cgr1000 c800 ir800 | The AAA server returned an error or was unreachable while attempting to authenticate a meter. |
| act2lFailure | cgr1000 c800 ir800 | The system rebooted after a hardware ACT2 failure, and the ACT2 process is being invoked. |
| archiveLogModeDisabled | db | Database archive log mode is disabled. Hot backups are not permitted. |
| archiveLogModeEnabled | db | Database archive log mode is enabled. |
| batteryFailure | cgr1000 c800 ir800 | The battery failed. |
| batteryLow | cgr1000 c800 ir800 | The battery charge is below the normal range. |
| batteryNormal | cgr1000 c800 ir800 | The battery charge is in the normal range. |
| bbuConfigFailure | cgr1000 c800 ir800 | The battery back up (BBU) configuration failed. |
| bbuFirmwareDownloadFailed | cgr1000 c800 ir800 | The BBU firmware download failed. |
| bbuFirmwareDownloadPassed | cgr1000 c800 ir800 | The BBU firmware download passed. |
| bbuFirmwareMismatchFound | cgr1000 c800 ir800 | A BBU firmware mismatch was found. |

| Option | Device | Description |
|---|---|---|
| bbuFirmwareUpgradeFailure | cgr1000<br>c800<br>ir800<br>ir500 | The BBU firmware upgrade failed. |
| bbuLockOut | ir500 | The BBU is locked out. |
| bbuPowerOff | cgr1000<br>c800<br>ir800<br>ir500 | The battery backup unit is not powered. |
| cd11IfRogueApDetectedNotif | ap800 | - |
| cd11IfStationSwitchOverNotif | ap800 | - |
| ciscoIetfDot11QosExtChangeNotif | ap800 | - |
| ciscoWlanVlanWepChangeNotif | ap800 | - |
| coldBoot | cgmesh<br>cgr1000<br>c800<br>ir800<br>ap800 | A cold boot occurred or a mesh node registered due to cold boot. |
| configPushed | cgr1000<br>c800<br>ir800 | (Industrial Operations Kit only) The group configuration pushed to CGRs. |
| configRollback | cgr1000<br>c800<br>ir800<br>ap800 | Configuration rollback required. |
| criticallyLowFRASpace | db | The Flash Recovery Area (FRA) free space is critically low. Run database backup immediately or risk IoT FND and database failure. |
| criticallyLowTableSpace | db | Database "USERS" table space is critically low. Contact your DBA immediately to add more space or risk IoT FND and database failure. |
| defaultRouteLost | cgmesh | The mesh node lost the default route. |

| Option | Device | Description |
|---|---|---|
| deviceAdded | cgr1000 c800 ir800 | (Industrial Operations Kit only) The EID of the new device. |
| deviceLocChanged | cgr1000 c800 ir800 | This event occurs when the GPS location changed in relation to the configured Interval and Distance thresholds. |
| deviceRemoved | cgr1000 c800 ir800 | (Industrial Operations Kit only) The EID of the removed device. |
| deviceUnknown | IoT FND | Unknown device detected by NMS. |
| doorClose | cgr1000 c800 ir800 | The device door was closed. |
| doorOpen | cgr1000 c800 ir800 | The device door is open. |
| dot11AuthenticateFail | ap800 | Dot11 authentication failed for the access point |
| dot11Deauthenticate | ap800 | Dot11 deauthentication frame was detected. |
| dot11Disassociate | ap800 | Dot11 disassociation frame was detected. |
| dot1xAuthFailed | cgmesh | Dot1x authentication failed for the meter. |
| dot1xAuthFailure | cgmesh | A Dot1x authentication failure was detected. |
| dot1xAuthFlood | cgmesh | A Dot1x authentication flood was detected. |
| dot1xReauth | cgmesh | Multiple attempts to send the mesh key to the meter failed. Re-authentication is in progress. |

| Option | Device | Description |
|---|---|---|
| down | asr1000 cgmesh IoT FND cgr1000 c800 ir800 db ir500 ap800 | The specified device is down. |
| hardwareInsertion | cgr1000 c800 ir800 | A new piece of hardware was inserted into the chassis. |
| hardwareRemoval | cgr1000 c800 ir800 | Hardware was removed from the chassis. |
| HSMdown | IoT FND | The Hardware Security Module (HSM) is down. |
| HSMup | IoT FND | The HSM is down. |
| interfaceDown | asr1000 c800 ir800 | The device interface is down. |
| interfaceUp | asr1000 c800 ir800 | The device interface is up. |
| linecardFailure | cgr1000 c800 ir800 | Linecard failure detected. |
| linePowerFailure | cgr1000 c800 ir800 | The line power supply for the device failed. This is different from the device being turned off, as happens in a power outage. |

| Option | Device | Description |
|--------|--------|-------------|
| linePowerRestored | cgr1000 c800 ir800 | The line power supply for the device is restored. |
| lowBattery | cgr1000 c800 ir800 ir500 | The device battery backup unit charge is low or below the specified threshold. |
| lowFlashSpace | cgr1000 c800 ir800 | The device is nearly out of memory on the flash partition. |
| lowFlashSpaceOk | cgr1000 c800 ir800 | The CGR available flash memory is within the specified threshold. |
| lowFRASpace | db | Low database CGR space detected. |
| lowMemory | cgr1000 c800 ir800 | Available memory is below the specified threshold. |
| lowMemoryOk | cgr1000 c800 ir800 | Available memory is above the specified threshold. |
| manualCloseEvent | asr1000 cgmesh cgr1000 c800 ir800 | The issue state changed by admin to closed. |
| manualNMSAddrChange | cgmesh | The mesh node registered due to a manual NMS address change. |
| manualReRegistration | cgmesh | The mesh node registered due to manual registration |
| meshConnectivityLost | cgmesh | The mesh node lost all connectivity. |
| meshLinkKeyTimeout | cgmesh | The mesh node link key timed out. |
| meshUpgradeSuccess | cgmesh | The mesh module firmware upgrade was successful. |

| Option | Device | Description |
|---|---|---|
| meterCertChange | cgmesh | The mesh node registered due to a certificate change. |
| metricRetrievalFailure | asr1000 cgr1000 c800 ir800 | Metric retrieval failed. |
| migratedToBetterPAN | cgmesh | The mesh node migrated to a better PAN. |
| modemTemperatureColdAlarm | cgr1000 c800 ir800 | The temperature of the modem module fell below specified levels. |
| modemTemperatureColdAlarmRecovery | cgr1000 c800 ir800 | The modem alarm reset. |
| modemTemperatureWarmAlarm | cgr1000 c800 ir800 | The temperature of the modem module fell above specified levels. |
| modemTemperatureWarmAlarmRecovery | cgr1000 c800 ir800 | The modem alarm reset. |
| nmsAddrChange | cgmesh | The mesh node registered due to an NMS address change. |
| nmsError | cgmesh | The mesh node registered due to an NMS error. |
| normalFRASpace | db | Database FRA space is normal. |
| normalTableSpace | db | Database table space is normal. |
| outage | cgmesh cgr1000 c800 ir800 ir500 | An outage was detected for this device. |
| portDown | cgr1000 c800 ir800 ap800 | Ethernet interface {0}/{1} is down. |

| Option | Device | Description |
|---|---|---|
| portFailure | cgr1000<br>c800<br>ir800<br>ap800 | Syslog message corresponding to the port facility was generated. |
| portUp | cgr1000<br>c800<br>ir800<br>ap800 | Ethernet interface {0}/{1} is up. |
| powerSourceNormal | cgr1000<br>c800<br>ir800 | The input power source is equal to or better than when system started. |
| powerSourceWarning | cgr1000<br>c800<br>ir800 | One or more input power source is not connected. |
| refreshMeshKeyFailed | asr1000<br>cgmesh | A refresh of the mesh key failed. |
| registered | asr1000<br>cgmesh | The event received is registered with NMS. |
| registrationFailure | cgr1000<br>c800<br>ir800<br>ap800 | Device registration failed. |
| registrationRequest | cgr1000<br>c800<br>ir800<br>ap800 | A registration request from a device was received. |
| registrationSuccess | cgr1000<br>c800<br>ir800<br>ap800 | Device registration was successful. |
| rejoinedWithNewIP | cgmesh | The mesh node registered with a new IP address. |

| Option | Device | Description |
|---|---|---|
| restoration | cgmesh | The device was restored from outage. |
| restorationRegistration | cgmesh | The mesh node registered after an outage. |
| rplTreeReset | cgr1000 c800 ir800 | The RPL tree version was reset to 2. Because the RPL tree is updated with data, versions increment. A value of 2 signifies a reset to its original initial state. |
| rplTreeSizeCritical | cgr1000 c800 ir800 | More than the maximum number of mesh nodes joined the RPL tree. |
| rplTreeSizeCriticalClear | cgr1000 c800 ir800 | Less than the maximum number of mesh nodes detected in the RPL tree. |
| rplTreeSizeMajor | cgr1000 c800 ir800 | More than the expected mesh nodes joined the RPL tree were detected. |
| rplTreeSizeMajorClear | cgr1000 c800 ir800 | The expected number of mesh nodes in the RPL tree were detected. |
| rplTreeVersionReset | cgr1000 c800 ir800 | The RPL tree version was reset to 2. Because the RPL tree is updated with data, versions increment. A value of 2 signifies a reset to its original initial state. |
| ruleEvent | asr1000 cgmesh IoT FND cgr1000 c800 ir800 db | This is a rule-generated event. |
| sdcardRemovalAlarm | cgr1000 | SD card removal detected, and an alarm sent. |

| Option | Device | Description |
|---|---|---|
| signatureFailure | asr1000<br><br>cgmesh<br><br>cgr1000<br><br>IoT<br>FND<br><br>c800<br><br>ir800<br><br>db<br><br>ir500 | Invalid signature reported by mesh nodes.<br><br>If this event occurs, you must verify the certificate setup and that the mesh node and IoT FND are time synchronized. |
| softwareCrash | cgr1000<br><br>c800<br><br>ir800 | Software process failed with a stateless restart, indicating an interruption of a service. Messages are processed for crashes on supervisor modules and line cards. |
| systemSwInconsistent | cgr1000<br><br>c800<br><br>ir800 | Inconsistency detected in software or file system. |
| temperatureMajorAlarm | cgr1000<br><br>c800<br><br>ir800 | A thermal sensor indicates that the temperature has reached the operating major threshold. |
| temperatureMajorRecovery | cgr1000<br><br>c800<br><br>ir800 | A major temperature alarm has recovered. |
| temperatureMinor | cgr1000<br><br>c800<br><br>ir800 | The device temperature reached the minor threshold. |
| temperatureMinorRecovery | cgr1000<br><br>c800<br><br>ir800 | The minor temperature alarm has recovered. |
| timeMismatch | cgmesh | The NMS server time does not match the device local time. |
| timeMismatchResolved | cgmesh | The NMS server time matches the device local time. |
| tunnelDown | cgr1000<br><br>c800<br><br>ir800 | The tunnel is down. |

| Option | Device | Description |
|---|---|---|
| tunnelProvFailure | cgr1000 c800 ir800 | Tunnel provisioning failed. |
| tunnelProvRequest | cgr1000 c800 ir800 | A tunnel provisioning request was received from a device. |
| tunnelProvSuccess | cgr1000 c800 ir800 | Tunnel provisioning was successful. |
| tunnelUp | cgr1000 c800 ir800 | The tunnel is up. |
| unknown | asr1000 cgmesh IoT FND cgr1000 c800 ir800 ir500 ap800 db | The event received is not registered with NMS. |
| unknownRegReason | cgmesh | The mesh node registered for an unknown reason. |
| unknownWPANChange | cgmesh | The mesh node changed its WPAN for an unknown reason. |
| unsupported | cgr1000 c800 ir800 ap800 | Unsupported device detected. |

| Option | Device | Description |
|--------|--------|-------------|
| up | asr1000 cgmesh IoT FND cgr1000 c800 ir800 ap800 db | The specified device is up. |
| veryLowFRASpace | db | Very low database FRA space detected. |
| wpanWatchdogReload | cgr1000 c800 ir800 | The WatchDog reloaded the WPAN module. The bridge was unresponsive for more than 5 minutes and the WatchDog is enabled. |

### Results

*Table 4: searchEvents Results*

| Field | Type | Description |
|-------|------|-------------|
| subscriptionid | long | Subscription ID used by the listener to identify the subscription response origin. |
| events | List<EventDetail> | Details about the event. |

### searchEvents SOAP XML Request Format

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:even="http://event.nbapi.cgms.cisco.com/">
   <soapenv:Header/>
   <soapenv:Body>
      <even:searchEvents>
         <!--Optional:-->
         <query>deviceType:cgmesh eventName:up</query>
         <!--Optional:-->
         <count>4</count>
         <!--Optional:-->
         <offset>0</offset>
      </even:searchEvents>
   </soapenv:Body>
</soapenv:Envelope>
```

# subscribeForEvents

This call streams a set of events to the API listener based on the query. Event subscriptions are based on device type, event name, or severity. Listener registers the URL and specifies the push window. After every configured eventPushWindowSec event push window, all new events that are received for the subscribed domain in this window are delivered to the registered URL. Subscription-based events notification uses the same query language as searchEvents, on page 1, except that the eventTime attribute-based queries cannot be subscribed to and return unsuccessful subscription errors.

> **Note** You can ONLY subscribe or unsubscribe to events of the domain that you belong to. Only root users have the privilege to subscribe or unsubscribe to events of other domains.

### Prototype

```
<even:subscribeForEvents>
<soapEndPointUrl>http://customer.network.com:11001/Process/Service/ProcessCellRouterStates/ReceiveEvents/EventPushService?wsdl</soapEndPointUrl>

<query>deviceType:cgmesh eventName:registered</query>
<eventPushWindowSec>21</eventPushWindowSec>
<domain>root</domain>
</even:subscribeForEvents>
```

### Parameters

*Table 5: subscribeForEvents Parameters*

| Parameter | Type | Description |
|---|---|---|
| soapEndPointUrl | string | The address of the WSDL file–as implemented by your client–that receives event notifications from the IoT FND NB API. For more information, see Handling Event Notifications On the Client Side, on page 21.<br><br>http://*<server_address>* :*<port number>* /*<path><api>* ?wsdl<br><br>For example:<br><br>http://localhost:8445/event?wsdl<br><br>**soapEndPointUrl** must point to the WSDL document that describes the listener application receiving the events. Ensure that the *target namespace* and *service name* parameters in the Web Services Description Language (WSDL) document conform to these default values:<br><br>```<br>target namespace<br>"http://pushevent.nbapi.cgms.cisco.com/"<br>service name<br>"EventPushService"<br>```<br><br>**Note**      Release 2.1 and later installations support https for soapEndPointUrl.<br><br>**Functionality in IoT FND 1.1.3 and later installations**<br><br>Some applications cannot set the default values for these parameters when generating the WSDL file. If this is the case, in IoT FND 1.1.3 and later installations you can set the following properties in the server/cgms/conf/cgms.properties file to match the values in the generated WSDL document:<br><br>```<br>eventSubscriberNamespace<br>="http://event.nbapi.cgms.mydomain.com/"<br>eventSubscriberServicename<br>="MyEventService"<br>```<br><br>The *target namespace* and *service name* parameters must match those specified in the cgms.properties file. If they are not specified in the cgms.properties file, they must match the default values.<br><br>**Note**      Only one set of these values is allowed in the cgms.properties file. If there are multiple subscribers, they must use the *target namespace* and *service name* parameter values specified in all WSDL documents pointed to in the **soapEndPointUrl** of the subscriptions. |
| query | string | The query string.<br><br>**Note**      The query language eventTime field cannot be inside the subscription. |
| eventPushWindowSec | integer | The event push window time, in seconds. The query executes after x seconds, and the results are pushed to the listener endpoint specified in the **soapEndPointUrl** WSDL file. |
| Domain | string | Specify the domain name to subscribe to events. |

**Results**

*Table 6: subscribeforEvents Response*

| Field | Type | Description |
|---|---|---|
| subscriptionId | long | Subscription ID |

The QueryResult Status field indicates if the subscription succeeded or failed.

**subscribeForEvents SOAP XML Request Format**

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:even="http://event.nbapi.cgms.cisco.com/">
   <soapenv:Header/>
   <soapenv:Body>
      <even:subscribeForEvents>
         <!--Optional:-->
         <soapEndPointUrl>http://localhost</soapEndPointUrl>
         <!--Optional:-->
         <query>deviceType:cgmesh eventName:registered</query>
        <!--Optional:-->
         <eventPushWindowSec>60</eventPushWindowSec>
         <!--Mandatory field for new subscriptions from 4.9.0-->
         <!--To update subscriptions prior to 4.9.0, this field should be empty-->
         <domain>root</domain>
      </even:subscribeForEvents>
   </soapenv:Body>
</soapenv:Envelope>
```

# unSubscribeForEvents

This call unsubscribes the defined listener event query.

**Prototype**

```
<even:unSubscribeForEvents>
<query>deviceType:cgmesh eventName:registered</query>
<domain>root</domain>
</even:unSubscribeForEvents>
```

**Parameters**

*Table 7: unSubscribeForEvents Parameters*

| Parameter | Type | Description |
|---|---|---|
| soapEndPointUrl | string | Address where the EventNbapiService WSDL is located. http://*<server_address>* :*<port number>* /*<path><api>* ?wsdl |
| query | string | Query string. **Note** Query language eventTime field cannot be inside the subscription. |

| Parameter | Type | Description |
|---|---|---|
| Domain | string | Specify the domain name to unsubscribe to events. |

### Results

*Table 8: unSubscribeForEvents Results*

| Field | Type | Description |
|---|---|---|
| subscriptionId | long | Subscription ID |

The QueryResult Status field indicates if the subscription succeeded or failed.

### unSubscribeForEvents SOAP XML Request Format

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:even="http://event.nbapi.cgms.cisco.com/">
   <soapenv:Header/>
   <soapenv:Body>
      <even:unSubscribeForEvents>
         <!--Optional:-->
         <soapEndPointUrl>http://localhost</soapEndPointUrl>
         <!--Optional:-->
         <query>deviceType:cgmesh eventName:registered</query>
         <!--Mandatory field for new subscriptions from 4.9.0-->
         <!--To update subscriptions prior to 4.9.0, this field should be empty-->
         <domain>root</domain>
      </even:unSubscribeForEvents>
   </soapenv:Body>
</soapenv:Envelope>
```

# subscribeForCgmeshOutage

This call is similar to subscribeForEvents, except that it is for outage and restoration events. Up to 10 subscribers (listeners) at a time can register for these events.

### Prototype

```
<even:subscribeForCgmeshOutage>
<soapEndPointUrl>http://128.107.109.98:8456/nbapi/pushevent?wsdl</soapEndPointUrl>
<domain>root</domain>
</even:subscribeForCgmeshOutage>
```

**subscribeForCgmeshOutage**

### Parameters

*Table 9: subscribeForCgmeshOutage Parameters*

| Parameter | Type | Description |
|---|---|---|
| soapEndPointUrl | string | Address of the WSDL, implemented by your client that receives outage notifications from the IoT FND NB API.<br><br>For more information, see Handling Event Notifications On the Client Side, on page 21.<br><br>http://*<server_address>* :*<port number>* /*<path><api>* ?wsdl<br><br>For example:<br><br>http://localhost:8445/outage?wsdl<br><br>**Note**     Release 2.1 and later installations support https for soapEndPointUrl. |
| Domain | string | Specify the domain name to subscribe to cgmesh outage events. |

To configure the amount of time, in seconds, after which IoT FND pushes batches of outage events and Restoration Events to all subscribers, set the value of the event-Outage-push-sec parameter in the /opt/cgms/conf/cgms.properties file. For example, to set event-Outage-push-sec to 30, add this line to the file:

```
event-Outage-push-sec=30
```

When IoT FND pushes outage events to subscribers, only subscribers that are up receive the events. The subscribers that are down (they do not respond) do not receive these events even after they come back online, but they receive the next outage event push.

For a very fast outage event push, set event-Outage-push-sec to a value as low as one second. If you set the push value to one second, IoT FND executes a job to find and push new events in the queue.

To prevent excessive alerts of momentary outages, set the value in the event-Outage-BufferTime parameter in cgms.properties file. The default value is 30 seconds. This value defines the duration during which IoT FND suppresses an outage notification before processing it.

The allowed-outage-skew is the acceptable time difference between the outage timestamp and restoration timestamp of the cgmesh and the default value for the skew is 1000 ms (configured in granularity of ms). When there is an outage, IoT FND receives Power Outage Notification (PON) which has the outage time and when cgmesh is restored in the network IoT FND receives Power Restoration Notification (PRN) which has the outage time and the restoration time. When the outage time stamp is different in PON and PRN, IoT FND processes the most recent PRN. The outage timestamps within this skew are treated as same outage event.

### Results

*Table 10: Subscribe for CG-Mesh Outage Response*

| Parameter | Type | Description |
|---|---|---|
| subscriptionId | long | Subscription ID which can be used by the listener to identify for which subscription they are getting the response from. |

The QueryResult Status field indicates if the subscription succeeded or failed.

### subscribeForCgmeshOutage SOAP XML Request Format

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:even="http://event.nbapi.cgms.cisco.com/">
   <soapenv:Header/>
   <soapenv:Body>
      <even:subscribeForCgmeshOutage>
         <!--Optional:-->
         <soapEndPointUrl>http:/localhost</soapEndPointUrl>
         <!--Mandatory field for new subscriptions from 4.9.0 -->
         <!-- To update subscriptions prior to 4.9.0, this field should be empty-->
         <domain>root</domain>
      </even:subscribeForCgmeshOutage>
   </soapenv:Body>
</soapenv:Envelope>
```

# unSubscribeForCgmeshOutage

This call unsubscribes the defined listener.

### Prototype

```
<even:unSubscribeForCgmeshOutage>
<soapEndPointUrl>http://128.107.109.98:8456/nbapi/pushevent?wsdl</soapEndPointUrl>
<domain>root</domain>
</even:unSubscribeForCgmeshOutage>
```

### Parameters

The following table describes the parameters in the request.

**Table 11: unSubscribeForCgmeshOutage Parameters**

| Parameter | Type | Description |
|---|---|---|
| soapEndPointUrl | string | Address of the Event WSDL service. The soapEndPointUrl identifies the subscription. |
| | | http://*<server_address>* :*<port number>* /*<path><api>* ?wsdl |
| | | **Note**        Release 2.1 and later installations support HTTPS for soapEndPointUrl. |
| Domain | string | Specify the domain name to unsubscribe to cgmesh outage events. |

### Results

**Table 12: unSubscribeForCgmeshOutage Results**

| Parameter | Type | Description |
|---|---|---|
| subscriptionId | long | Subscription ID |

The QueryResult Status field indicates if the subscription succeeded or failed.

### unSubscribeForCgmeshOutage SOAP XML Request Format

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:even="http://event.nbapi.cgms.cisco.com/">
   <soapenv:Header/>
   <soapenv:Body>
      <even:unSubscribeForCgmeshOutage>
         <!--Optional:-->
         <soapEndPointUrl>http:/localhost</soapEndPointUrl>
         <!--Mandatory field for new subscriptions from 4.9.0-->
         <!--To update subscriptions prior to 4.9.0, this field should be empty-->
         <domain>root</domain>
      </even:unSubscribeForCgmeshOutage>
   </soapenv:Body>
</soapenv:Envelope>
```

# getAllEventSubscriptions

This API gets all the events or outage events for which you have subscribed in IoT FND.

Based on the value defined in the `outageSubscription` field, the API provides information on either all events or only outage events:

- Set the value to "`true`" to get information on all the outage events.

- Set the value to "`false`" to get information on all the events.

### Prototype

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:even="http://event.nbapi.cgms.cisco.com/">
   <soapenv:Header/>
   <soapenv:Body>
      <even:getAllEventSubscriptions>
         <!--Optional:-->
         <outageSubscription>false</outageSubscription>
          </even:getAllEventSubscriptions>
   </soapenv:Body>
</soapenv:Envelope>
```

### Parameters

*Table 13: getAllEventSubscriptions Parameters*

| Parameter | Type | Description |
|---|---|---|
| outageSubscription | boolean | • Set the value to "true" to get the outage events.<br><br>• Set the value to false to get all events. |
| Domain | string | Specify the domain name to subscribe to all events or only outage events. |

### getAllEventSubscriptions SOAP XML Request Format

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Header xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"/>
   <soap:Body>
```

```
            <ns2:getAllEventSubscriptionsResponse xmlns:ns2="http://event.nbapi.cgms.cisco.com/">

        <eventSubscriptionListResult>
            <queryStatus>SUCCEEDED</queryStatus>
            <eventSubscriptions>
                <eventPushWindowSec>60</eventPushWindowSec>
                <id>1000</id>
                <isOutageSubscription>false</isOutageSubscription>
                <lastPushTime>0</lastPushTime>
                <queryString>deviceType:ir800</queryString>
                <soapEndPointUrl>http:/localhost</soapEndPointUrl>
                <domain>root</domain>
            </eventSubscriptions>
            <numberOfRecords>1</numberOfRecords>
            <showOutageSubscriptionOnly>false</showOutageSubscriptionOnly>
        </eventSubscriptionListResult>
    </ns2:getAllEventSubscriptionsResponse>
  </soap:Body>
</soap:Envelope>
```
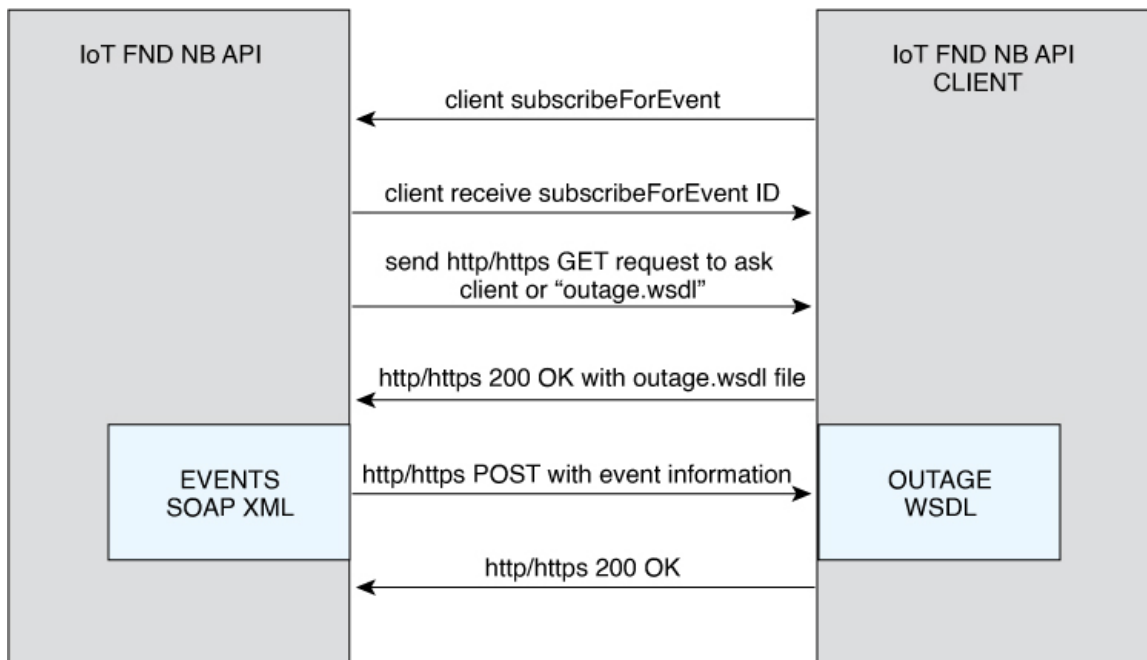
# Handling Event Notifications On the Client Side

When subscribing for an event type, your IoT FND NB API client must implement a Web Service that implements the WSDL for handling event notifications sent by the IoT FND NB API. The WSDL you must provide the receiveEvents() method, which the IoT FND NB API uses to send event notifications to your client.

## Example

In the following figure, the IoT FND NB API client implements the Outage WSDL. When the client subscribes for outage events, IoT FND uses the IoT FND NB API to call the method receiveEvents() on the IoT FND NB API client.

**Figure 1: Event Notification Handling**



This is the Event XML that your client must implement to receive outage notifications from IoT FND.

# Events and Outage Notification Handling WSDL (Client Side)

The clients must implement the WSDL file to receive notification on events and power outages. The WSDL file varies for different releases of IoT FND.

- For IoT FND releases prior to 4.9.x, the WSDL file  is same for both events and outage notifications.

- For IoT FND release 4.9.x to 4.10.x, the WSDL file is different for events and outage notifications.

  - Events

  - Outages

**WSDL File for Event and Outage Notification**

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
 xmlns:tns="http://pushevent.nbapi.cgms.cisco.com/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema" name="eventPush"
 targetNamespace="http://pushevent.nbapi.cgms.cisco.com/">
 <wsdl:types>
  <xsd:schema targetNamespace="http://pushevent.nbapi.cgms.cisco.com/">
   <xsd:element name="receiveEvents" type="tns:receiveEvents" />
   <xsd:complexType name="receiveEvents">
    <xsd:sequence>
     <xsd:element minOccurs="0" name="eventQueryResult"
      type="tns:eventQueryResult" />
    </xsd:sequence>
   </xsd:complexType>
```

```
<xsd:complexType name="eventQueryResult">
 <xsd:complexContent>
  <xsd:extension base="tns:queryResult">
   <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
     name="events" nillable="true" type="tns:eventDetail" />
    <xsd:element name="subscriptionId" type="xsd:long" />
   </xsd:sequence>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
<xsd:complexType abstract="true" name="queryResult">
 <xsd:sequence>
  <xsd:element minOccurs="0" name="queryId" type="xsd:string" />
  <xsd:element minOccurs="0" name="queryStatus" type="xsd:string" />
 </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="eventDetail">
 <xsd:sequence>
  <xsd:element minOccurs="0" name="eid" type="xsd:string" />
  <xsd:element minOccurs="0" name="eventMessage" type="xsd:string" />
  <xsd:element minOccurs="0" name="eventSeverity" type="xsd:string" />
  <xsd:element minOccurs="0" name="eventTime" type="xsd:long" />
  <xsd:element minOccurs="0" name="eventTypeName" type="xsd:string" />
  <xsd:element minOccurs="0" name="meterId" type="xsd:string" />
 </xsd:sequence>
</xsd:complexType>
</xsd:schema>
</wsdl:types>
<wsdl:message name="receiveEvents">
 <wsdl:part element="tns:receiveEvents" name="receiveEvents" />
</wsdl:message>
<wsdl:portType name="EventPushService">
 <wsdl:operation name="receiveEvents">
  <wsdl:input message="tns:receiveEvents" />
 </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="EventPushServiceBinding" type="tns:EventPushService">
 <soap:binding style="document"
  transport="http://schemas.xmlsoap.org/soap/http" />
 <wsdl:operation name="receiveEvents">
  <soap:operation soapAction="http://pushevent.nbapi.cgms.cisco.com/receiveEvents" />
  <wsdl:input>
   <soap:body use="literal" />
  </wsdl:input>
 </wsdl:operation>
</wsdl:binding>
<wsdl:service name="EventPushService">
 <wsdl:port binding="tns:EventPushServiceBinding" name="EventPushService">
  <soap:address location="http://127.0.0.1:8008" />
 </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

**WSDL File for Outage Notification**

✎

**Note**    From IoT FND release 4.9.x to 4.10.x, the domain is specified in the request.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<wsdl:definitions
 xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
```

```
xmlns:tns="http://pushevent.nbapi.cgms.cisco.com/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" name="eventPush"
targetNamespace="http://pushevent.nbapi.cgms.cisco.com/">
<wsdl:types>
 <xsd:schema targetNamespace="http://pushevent.nbapi.cgms.cisco.com/">
  <xsd:element name="receiveEvents" type="tns:receiveEvents" />
  <xsd:complexType name="receiveEvents">
   <xsd:sequence>
    <xsd:element minOccurs="0" name="eventQueryResult" type="tns:eventQueryResult" />
   </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="eventQueryResult">
   <xsd:complexContent>
    <xsd:extension base="tns:queryResult">
     <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="events" nillable="true"
type="tns:eventDetail" />
      <xsd:element name="subscriptionId" type="xsd:long" />
     </xsd:sequence>
    </xsd:extension>
   </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType abstract="true" name="queryResult">
   <xsd:sequence>
    <xsd:element minOccurs="0" name="queryId" type="xsd:string" />
    <xsd:element minOccurs="0" name="queryStatus" type="xsd:string" />
   </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="eventDetail">
   <xsd:sequence>
    <xsd:element minOccurs="0" name="domain" type="tns:domain"/>
    <xsd:element minOccurs="0" name="eid" type="xsd:string" />
    <xsd:element minOccurs="0" name="eventMessage" type="xsd:string" />
    <xsd:element minOccurs="0" name="eventSeverity" type="xsd:string" />
    <xsd:element minOccurs="0" name="eventTime" type="xsd:long" />
    <xsd:element minOccurs="0" name="eventTypeName" type="xsd:string" />
    <xsd:element minOccurs="0" name="meterId" type="xsd:string" />
   </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="domain">
   <xsd:sequence>
    <xsd:element minOccurs="0" name="id" type="xsd:long"/>
    <xsd:element minOccurs="0" name="name" type="xsd:string"/>
   </xsd:sequence>
  </xsd:complexType>
 </xsd:schema>
</wsdl:types>
<wsdl:message name="receiveEvents">
 <wsdl:part element="tns:receiveEvents" name="receiveEvents" />
</wsdl:message>
<wsdl:portType name="EventPushService">
 <wsdl:operation name="receiveEvents">
  <wsdl:input message="tns:receiveEvents" />
 </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="EventPushServiceBinding" type="tns:EventPushService">
 <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
 <wsdl:operation name="receiveEvents">
  <soap:operation soapAction="http://pushevent.nbapi.cgms.cisco.com/receiveEvents" />
  <wsdl:input>
   <soap:body use="literal" />
  </wsdl:input>
 </wsdl:operation>
```

```
      </wsdl:binding>
      <wsdl:service name="EventPushService">
       <wsdl:port binding="tns:EventPushServiceBinding" name="EventPushService">
        <soap:address location="http://10.88.202.38:33005/FNDIncomingEvent" />
       </wsdl:port>
      </wsdl:service>
     </wsdl:definitions>
```

# Push Mechanisms

Push mechanisms work only when the NMS server has successfully completed the subscription, as defined in the subscribeForEvents, on page 14 and subscribeForCgmeshOutage, on page 17 API methods.

A successful subscription leads to generation of the Subscription ID that is sent to the subscriber. The subscriber uses the Subscription ID to track the event push.

IoT FND runs a Scheduled Job every x seconds. Seconds are configurable during the subscription by using the event-Outage-push-sec global parameter defined in the /opt/cgms/conf/cgms.properties file. After every x seconds, IoT FND generates an EventList and pushes it to the subscribers defined in the soapEndPointUrl.

The web service to implement on the NMS side is:

```
public void receiveEvents(EventQueryResult eventQueryResult) throws java.rmi.RemoteException;
```

## QueryResult

The QueryResult Status field indicates if the subscription succeeded or failed. The following table describes the parameters in the response.

**Table 14: EventQueryResult Response**

| Field | Type | Description |
|---|---|---|
| subscriptionId | long | Subscription ID used by the listener to identify which subscription the response is from. |
| events | List<*EventDetail*> | Details about the event. |

The following table describes the parameters in the EventDetail results.

**Table 15: EventDetail Response**

| Field | Type | Description |
|---|---|---|
| eid | string | Serial number for the CGR and MAC address for the mesh endpoint. |
| eventMessage | string | Message related to the event. |
| eventTime | long | Time in milliseconds. |
| eventTypeName | string | Type of event |
| meterId | string | Meter ID corresponding to the EID. For this value to return, it must be included in the import file when importing meters in IoT FND. |

# Example

This is an example of the XML content that the subscriber receives:

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
   <env:Header>
      <seam:conversationId
xmlns:seam="http://www.jboss.org/seam/webservice">110</seam:conversationId>
   </env:Header>
   <env:Body>
      <ns2:searchEventsResponse xmlns:ns2="http://event.nbapi.cgms.cisco.com/">
         <eventQueryResult>
            <queryId></queryId>
            <queryStatus>SUCCEEDED</queryStatus>
            <events>
               <eid>NE01</eid>
               <eventMessage>Device is Up</eventMessage>
               <eventSeverity>INFO</eventSeverity>
               <eventTime>1314656731899</eventTime>
               <eventTypeName>up</eventTypeName>
               <meterId>Sjc123</meterId>
            </events>
            <events>
               <eid>NE01</eid>
               <eventMessage>Outage detected on this device</eventMessage>
               <eventSeverity>CRITICAL</eventSeverity>
               <eventTime>1314656731908</eventTime>
               <eventTypeName>outage</eventTypeName>
               <meterId>Sjc123</meterId>
            </events>
            <events>
               <eid>NE01</eid>
               <eventMessage>Device has been Restored from Outage</eventMessage>
               <eventSeverity>INFO</eventSeverity>
               <eventTime>1314656771923</eventTime>
               <eventTypeName>restoration</eventTypeName>
               <meterId>Sjc123</meterId>
            </events>
            <events>
               <eid>NE01</eid>
               <eventMessage>Device is Up</eventMessage>
               <eventSeverity>INFO</eventSeverity>
               <eventTime>1314656771933</eventTime>
               <eventTypeName>up</eventTypeName>
               <meterId>Sjc123</meterId>
            </events>
            <subscriptionId>2</subscriptionId>
         </eventQueryResult>
      </ns2:searchEventsResponse>
   </env:Body>
</env:Envelope>
```

The subscriptionId XML element (<subscriptionId>2</subscriptionId>) tells the receiver that this push is for the subscription ID equal to 2.