C H A P T E R **6**

Configuring Event Notifications

Event notification is a feature that enables you to define conditions that cause the location server to send notifications to the listeners that you have specified in Cisco WCS. This chapter describes how to define events and event groups, and how to configure event notification parameters. It also describes how to view event notification summaries.

This chapter contains the following sections:

Working with Event Groups

This section describes how to add and delete event groups.

Adding Event Groups

To manage events more efficiently, you can use Cisco WCS to create event groups. Event groups help you organize your event definitions.

To add an event group, follow these steps:

Step 1   In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2   Choose **Notification Settings** (left- panel).

Step 3   From the Select a command drop-down menu, select **Add Event Group**, Click **Go**.

Step 4   Enter the name of the group in the Group Name field.

Step 5   Click **Save**.

The new event group appears in the Event Settings window.

Deleting Event Groups

To delete an event group, follow these steps:

Step 1   In Cisco WCS, choose **Services > Context Aware Notifications**

Step 2   Select the event group to delete by checking its corresponding check box.

Step 3   From the Select a command drop-down menu, select **Delete Event Group(s)**, and click **Go**.

Step 4   In the panel that appears, click **OK** to confirm deletion.

Step 5   Click **Save**.

Adding, Deleting, and Testing Event Definitions

An event definition contains information about the condition that caused the event, the assets to which the event applies, and the event notification destinations. This section describes how to add, delete, and test event definitions.

Adding an Event Definition

Cisco WCS enables you to add definitions on a per-group basis. Any new event definition must belong to a particular group.

To add an event definition, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications.**

Step 2 Choose **Settings** (left panel).

Step 3 Click the name of the group to which you want to add the event. An event settings summary window appears for the specific event group listing all defined event definitions.

Step 4 From the Select a command drop-down menu, select **Add Event Definition** and click **GO**.

Step 5 Enter a name for the event definition and click **Save**.

Step 6 At the Conditions tab, add one or more conditions. For each condition you add, specify the rules for triggering an event notification.

For example, to keep track of heart monitors in a hospital, you can add three rules to generate an event notification if the heart monitor is missing for two hours, if the heart monitor moves out of the second floor, or if the heart monitor enters a specific coverage area within a floor.

To add a condition, follow these steps:

a. Click **Add** to add a condition that triggers this event.

b. In the Add/Edit Condition dialog box, follow these steps:

    1. Choose a condition type from the Condition Type drop-down menu.

    2. In the Trigger If field, follow these steps:

    If you chose **Missing** from the Condition Type drop-down menu, enter the number of minutes after which a missing asset event is generated. For example, if you enter 10 in this field, the location server generates a missing asset event if the location server has not located the asset for more than 10 minutes. Proceed to Step c.

    If you chose **In/Out** from the Condition Type drop-down menu, select **Inside of** or **Outside of**, then click **Select Area** to select the area to monitor for assets going into it or out of it. In the Select dialog box, choose the area to monitor, then click **Select**. The area to monitor could be an entire campus, building within a campus, a floor in a building, or a coverage area (you can define a coverage area using the map editor). For example, to monitor part of a floor in a building, choose a campus from the Campus drop-down menu, choose a building from the Building drop-down menu, and choose the area to monitor from the Floor Area drop-down menu. Then click **Select**. Proceed to Step c.

    If you chose **Distance** from the Condition Type drop-down menu, enter the distance in feet that will trigger an event notification if the monitored asset moves beyond the specified distance from a designated marker, then click **Select Marker**. In the Select dialog box, select the campus, building, floor, and marker from the corresponding drop-down menus and click **Select**. For example, if you add a marker to a floor plan and set the distance in the Trigger If field to 60 feet, an event notification will be generated if the monitored asset moves 73 feet away from the marker. Proceed to Step c.

✎

Note You can create markers and coverage areas using the Map Editor. When you create marker names, make sure they are unique across the entire system.

If you chose **Battery Level** from the Condition Type drop-down menu, check the box next to the appropriate battery level (low, medium, normal) that will trigger an event. Proceed to Step c.

If you chose **Location Chang**e from the Condition Type drop-down menu, proceed to Step c.

If you chose **Emergency** from the Condition Type drop-down menu, click the button next to the appropriate emergency (any, panic button, tampered, detached, unknown) that will trigger an event. Proceed to Step c.

If you chose **Chokepoint** from the Condition Type drop-down menu. There is only one trigger condition and it is displayed by default. No configuration is required. Proceed to Step c.

> **Note**   For chokepoints you must select the chokepoint after you add the condition. See directions in the note after sub-step e.

c. From the Apply To drop-down menu, choose the type of asset (Any, Clients, Tags, Rogue APs, or Rogue Clients) for which an event will be generated if the trigger condition is met.

> **Note**   Emergency and chokepoint events apply only to tags (Cisco CX v.1 compliant).

d. From the Match By drop-down menu, choose the matching criteria (MAC Address, Asset Name, Asset Group, or Asset Category), the operator (**Equals** or **Like**) from the drop-down menu, and enter the relevant text for the selected Match By element.

Following are examples of asset matching criteria that you can specify:

– If you choose **MAC Address** from the Match By drop-down menu, choose **Equals** from the Operator drop-down menu, and enter **12:12:12:12:12:12**, the event condition applies to the element whose MAC address is 12:12:12:12:12:12 (exact match).

– If you choose **MAC Address** from the Match By drop-down menu, choose **Like** from the Operator drop-down menu, and enter **12:12**, the event condition applies to elements whose MAC address starts with 12:12.

e. Click **Add** to add the condition you have just defined.

> **Note**   If you are defining a chokepoint, you must select the chokepoint after you add the condition.

To select a chokepoint, do the following:

1. Click **Select Chokepoint**. An entry panel appears.

2. Select Campus, Building and Floor from the appropriate drop-down menus.

3. Select a Chokepoint from the menu that appears.

You are returned to the Add/Edit Condition panel and the location path (*Campus > Building > Floor*) for the chokepoint auto-populates the field next to the Select Checkpoint button.

Step 7   At the Destination and Transport tab, follow these steps to add one or more destinations to receive event notifications and configure the transport settings:

a. To add a new destination, click **Add New**.

b. Enter the IP address of the system that will receive event notifications, and click **OK**.

The recipient system must have an event listener running to process notifications. By default, when you create an event definition, Cisco WCS adds its IP address as the a destination.

    c. To select a destination to send event notifications to, highlight one or more IP addresses in the box on the right, and click **Select** to add the IP addresses to the box on the left.

    d. In the Message Format field, select **XML** or **Plain Text** to specify the message format.

       If you select WCS as the destination of event notifications, you must select the XML format.

    e. Choose one of the following transport types from the Transport Type drop-down menu:

       – **SOAP**—Specifies Simple Object Access Protocol, a simple XML protocol, as the transport type for sending event notifications. Use SOAP to send notifications over HTTP/HTTPS and to be processed by web services on the destination.

        If you choose **SOAP**, specify whether to send notifications over HTTPS by checking its corresponding check box. If you don't, HTTP is used. Also, enter the destination port number in the Port Number field.

       – **Mail**—Use this option to send notifications via email.

        If you choose **Mail**, you need to choose the protocol for sending the mail from the Mail Type drop-down menu. You also need to enter the following information: username and password (if Authentication is enabled), name of the sender, prefix to add to the subject line, email address of recipient, and a port number if necessary.

       – **SNMP**—Use Simple Network Management Protocol, a very common technology for network monitoring used to send notifications to SNMP-capable devices.

        If you choose **SNMP**, enter the SNMP community string in the SNMP Community field and the port number to send notifications to in the Port Number field.

       – **SysLog**—Specifies the system log on the destination system as the recipient of event notifications.

        If you choose **SysLog**, enter the notification priority in the Priority field, the name of the facility in the Facility field, and the port number on the destination system in the Port Number field.

    f. To enable HTTPS, check the **Enable** check box next to it.

    g. **Port Number** auto-populates when HTTPS is enabled.

    h. Click **Add**.

Step 8 Under the General tab, follow these steps:

    a. Enable event generation (disabled by default) by checking the **Enabled** check box for the Admin Status field.

    b. Set the event priority by choosing a number from the Priority drop-down menu. Zero is highest.

> **Note** An event definition with higher priority is serviced before event definitions with lower priority.

    c. Select the day(s) of the week you want to activate event notification by checking the box next to the day(s).

> **Note** If you want to continuously report events, select the **All the Time** option. In this case, there is no need to set start and end ranges for event notification. These options are not displayed.

    d. Select the time for starting the event notification by selecting the appropriate hour, minute and AM/PM options from the Apply From heading.

    e. Select the time for ending the event notification by selecting the appropriate hour, minute and AM/PM options from the Apply Until heading.

    f. Click **Save**.

Step 9 Verify that the new event definition is listed for the event group (Services > Notifications > Settings > *Event Group Name)*.

Deleting an Event Definition

To delete one or more event definitions from Cisco WCS, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2 Choose **Settings** (left panel).

Step 3 Click the name of the group from which you want to delete the event definition(s).

Step 4 Select the event definition that you want to delete by checking its corresponding check box.

Step 5 From the Select a command drop-down menu, choose **Delete Event Definition(s)**, and click **GO**.

Step 6 Click **OK** to confirm that you want to delete the selected event definition(s).

> **Note** Deleting event definitions as described above only removes them from the WCS database. You must also remove the definitions from the location server database.

To remove definitions from the location server, follow these steps:

Step 1 In Cisco WCS, choose **Services > Synchronize Services**.

Step 2 From the **Synchronize** drop-down menu, select **Event Groups**.

Step 3 To remove an event definition, click **Unassign** for the event group to which the event belongs.

> **Note** The Unassign link only appears if the event group is linked. When a group is already unassigned, the assign link appears next to the event group.

Step 4 Click **Synchronize**.

Testing Event Definitions

To verify that the location server is sending event definitions over the transport protocol you have specified in the event definition, use Cisco WCS to test the event notifications. The location server sends three fictitious event notifications (absence, containment, and distance) to the destinations you have specified in the event definition. The messages contain dummy MAC addresses.

> **Note**   Emergency and chokepoint event notifications are not tested.

To test one or more event definitions, follow these steps:

Step 1   In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2   Choose **Settings** (left panel).

Step 3   Click the name of the group containing the event definitions that you want to test.

Step 4   Select the event definitions that you want to test by checking their corresponding check boxes.

Step 5   From the Select a command drop-down menu, select **Test-Fire Event Definition(s)**, and click **GO**.

Step 6   Click **OK** to confirm that you want to test-fire event notifications.

Step 7   Check to make sure that notifications were sent to the designated recipient.

Viewing Event Notification Summary

The location server sends event notifications and does not store them. However, if WCS is a destination of notification events, it stores the notifications it receives and groups them into the following seven categories:

- Absence (Missing)—The location server generates absence events when the monitored assets go missing. In other words, the location server cannot detect the asset in the WLAN for the specified time.

- In/Out Area (Containment)—The location server generates containment events when an asset is moved inside or outside a designated area.

  > **Note**   You define a containment area (campus, building, or floor) in the Maps section of Cisco WCS (**Monitor > Maps**). You can define a coverage area using the Map Editor.

- Movement from Marker (Movement/Distance)—The location server generates movement events when an asset is moved beyond a specified distance from a designated marker you define on a map.

- Location Changes—The location server generates location change events when client stations, asset tags, rogue clients and rogue access points move from their previous location.

- Battery Level—The location server generates battery level events for all tracked asset tags.

- Emergency—The location server generates an emergency event for a CCX v.1 compliant asset tag when the tag's panic button is triggered or the tag becomes detached, tampered with, goes inactive or reports an unknown state. This information is only reported and displayed for CCX v.1 compliant tags.

- Chokepoint Notifications—The location server generates an event when a tag is seen (stimulated) by a chokepoint. This information is only reported and displayed for CCX v.1 compliant tags.

> **Note**   All element events are summarized hourly and daily.

To view event notifications, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Cisco WCS displays a summary of event notifications for each of the seven event notification categories.

> **Note** Emergency and chokepoint notifications are only reported and displayed for Cisco compatible CX v.1 compliant tags.

Step 2 To view event notifications for a monitored asset, click one of its corresponding links.

For example, to view absence events for client stations generated in the last hour, click the link in the Last Hour column for the Client Stations entry in the Absence (Missing) list.

Clicking one of these links searches for location notifications of all severities.

Notifications Cleared

A location server sends event notifications when it clears an event condition in one of the following scenarios:

- **Missing (Absence)**—Elements reappear.
- **In/Out Area (Containment)**—Elements move back in or out of the containment area.
- **Distance**—Elements move back within the specified distance from a marker.
- **Location Changes**—Clear state is not applicable to this condition.
- **Battery Level**—Tags are detected again operating with Normal battery level.

> **Note** In Cisco WCS, the Notifications Summary window reflects whether notifications for cleared event conditions have been received.

Enabling Notifications and Configuring Notification Parameters

Enabling Notifications

You can use Cisco WCS to define and enable user-configured conditional notifications and northbound notifications.

User-configured conditional notifications manage which notifications the mobility services engine sends to Cisco WCS. Refer to the "Adding, Deleting, and Testing Event Definitions" section on page 6-2.

Northbound notifications define which tag notifications the mobility services engine sends to third-party applications. Client notifications are not forwarded. By enabling northbound notifications in Cisco WCS, the following five event notifications are sent: chokepoints, telemetry, emergency, battery, and vendor data. To send a tag location, you must enable that notification separately.

The mobility services engine sends all northbound notifications in a set format. Details are available on the Cisco developers support portal at:

http://www.cisco.com/en/US/products/svcs/ps3034/ps5408/ps5418/serv_home.html

Filtering Northbound Notifications

Filtering on northbound notifications is possible in release 6.0 and later. Similar to user-configured conditional notifications, you can limit which event notifications are forwarded.

You can use filtering to focus on specific notifications important to tag monitoring within your network and to limit the overall number of notifications sent. The latter might preserve processing and storage capacity on the northbound platform.

Note   Cisco recommends defining northbound notification filters in the *aes-config.xml* file on the mobility services engine rather than Cisco WCS.

You can filter on six northbound parameters as summarized below:

```
<entry key="send-event-on-location-calc">true</entry>
<entry key="send-event-on-every-beacon">true</entry>
<entry key="send-event-on-vendor">true</entry>
<entry key="send-event-on-emergency">true</entry>
<entry key="send-event-on-chokepoint">true</entry>
<entry key="send-event-on-telemetry">true</entry>
```

To send all six northbound notifications with each beacon, ensure that the *send-event-on-location-calc* and *send-event-on-every-beacon* notification types are marked as true.

To limit the number of notifications, edit (but do not delete) the specific event entry in the *aes-config.xml* file by marking it as *false*.

For example, to send emergency and chokepoint notifications only change the other four notification types (location, beacon, vendor, and telemetry) to *false*.

The modified *aes-config.xml* file would read as:

```
<entry key="send-event-on-location-calc">false</entry>
<entry key="send-event-on-every-beacon">false</entry>
<entry key="send-event-on-vendor">false</entry>
<entry key="send-event-on-emergency">true</entry>
<entry key="send-event-on-chokepoint">true</entry>
<entry key="send-event-on-telemetry">false </entry>
```

Configuring Notification Parameters

You can limit the rate at which a location server generates notifications, set a maximum queue size for notifications, and set a retry limit for notifications with in a certain period.

Notification parameter settings apply to user-configurable conditional notifications and northbound notifications except as noted in Table 6-1.

Note   Modify notification parameters only when you expect the location server to send a large number of notifications or when notifications are not being received.

To enable northbound notifications and to configure notification parameters, follow these steps:

Step 1 In Cisco WCS, choose **Services > Mobility Services**.

Step 2 Click the name of the location server you want to configure.

Step 3 Choose **Context Aware Service > Advanced > Notification Parameters** to display the configuration options (see Figure 6-1).

Figure 6-1   *Advanced > Notification Parameters Window*
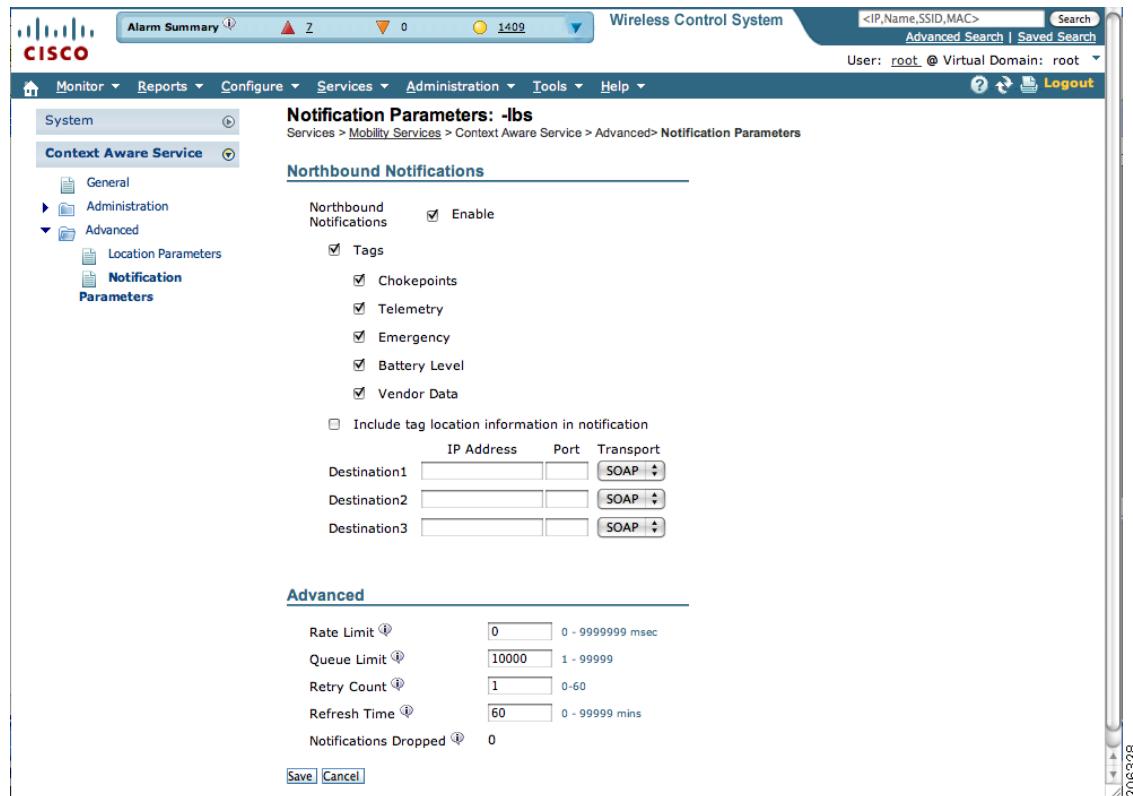
Step 4 Check the **Enable Northbound Notifications** check box to enable the function.

Step 5 Check the **Tags** check box to send tag notifications to third-party applications (northbound).

> ✎
>
> **Note**   To limit the types of northbound notifications sent for tags, edit the *aes-config.xml* file. Refer to the "Filtering Northbound Notifications" section on page 6-9.

Step 6 Check the **Include tag location information in notification** check box to send the tag location.

> ✎
>
> **Note**   You can define the type of location information to send for the tag. Options include building, X, Y map coordinates, civic (address, city, state), or GEO (longitude, latitude). Refer to the "Enabling Location Presence on a Location Server" section on page 7-38 section for configuration details.

Step 7 Enter the IP address and port for the system that is to receive the northbound notifications.

Step 8   Select the transport type from the drop-down menu.

Step 9   To modify the notification parameter settings, enter the new value in the appropriate field in the Advanced section of the window.

The notification parameters and their definitions are listed in Table 6-1.

Table 6-1   *Notification Parameters:*

Parameter	Description
Rate Limit	Enter the rate in milliseconds at which the location server generates notifications. A value of 0 (default) means that the mobility services engine generates notifications as fast as possible (Northbound notifications only).
Queue Limit	The event queue limit for sending notifications. The server will drop any event above this limit. Default value is 500.
Retry Limit	Enter the number of times to generate an event notification before the refresh time expires. This value ensures, to some extent, that the events that the location server generated will eventually reach WCS. Default value is 1. ✎ **Note** The location server does not store events in its database.
Refresh Time	Enter the wait time in minutes that must pass before an event notification is resent. For example, suppose you enter 30 in this field. If a monitored element goes out of a specified area, the location server sends an event notification. Then, until the event is cleared, the location server resends an event notification every 30 minutes.
Notifications Dropped	(Read only). The number of event notifications dropped from the queue since startup.

Step 10   Click **Save** to store your updates in the Cisco WCS and location server databases.

Notification Message Formats

This section describes the notification message formats.

Notification Formats in XML

This section describes the XML format of notification messages.

✎
Note   The XML format is part of a supported API and Cisco will provide change notification as part of the Location Server API program, whenever the API is updated in the future.

Missing (Absence) Condition

Message format for element absence:

```
<AbsenceTrackEvent
missingFor="<time in secs entity has been missing>"
lastSeen="time last seen"
trackDefn="<name of track definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for the clear state:

```
<AbsenceTrackEvent
state="clear"
trackDefn="<name of track definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Following are examples:

```
<AbsenceTrackEvent state="set" missingFor="34" lastSeen="15:00:20 28 May 2006"
trackDefn="absenceDef1" entityType="Mobile Station"
entityID="00:0c:f1:53:9e:c0"/>

<AbsenceTrackEvent state="clear" entityType="Tag"
trackDefn="absenceDef1" entityID="00:0c:cc:5b:fc:da"/>
```

In/Out (Containment) Condition

Message format for element containment:

```
<ContainmentTrackEvent
in="true | false"
trackDefn="<name of track definition>"
containerType="Floor | Area | Network Design | Building"
containerID="<fully quality name of container>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for the clear state:

```
<ContainmentTrackEvent
state="clear"
trackDefn="<name of track definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Following are examples:

```
<ContainmentTrackEvent in="true" trackDefn="myContainerRule1"
containerType="Area"
containerID="nycTestArea,5th Floor,Bldg-A,Rochester_Group,Rochester,"
entityType="Tag" entityID="00:0c:cc:5b:fa:44"/>
```

Note   The containerID string represents a coverage area called nycTestArea, located in the 5th floor of Bldg-A of the campus *Rochester*.

```
<ContainmentTrackEvent state="clear" entityType="Tag"
trackDefn="myContainerRule1" entityID="00:0c:cc:5b:f8:ab"/>
```

Distance Condition

Message format for elements in the same floor:

```
<MovementTrackEvent
distance="<distance in feet at which the element was located>"
triggerDistance="<the distance specified on the condition"
reference="<name of the marker specified on the condition>"
trackDefn="<name of event definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for elements located in a different floor:

```
<MovementTrackEvent optionMsg="has moved beyond original floor"
reference="<name of the marker specified on the condition>"
trackDefn="<name of event definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for clear state:

```
<MovementTrackEvent
state="clear"
trackDefn="<name of event definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Following are examples:

```
<MovementTrackEvent distance="115.73819627990147" triggerDistance="60.0"
reference="marker2" trackDefn="distance2" entityType="Mobile Station"
entityID="00:0c:41:15:99:92"/>

<MovementTrackEvent optionMsg="has moved beyond original floor"
reference="marker2" entityType="Tag"
trackDefn="distance2"
entityID="00:0c:cc:5b:fa:4c"/>

<MovementTrackEvent state="clear" entityType="Tag"
```

Battery Level

An example:

```
<BatteryLifeTrackEvent lastSeen="10:28:52 23 May 2006" batteryStatus="medium"
trackDefn="defn1" entityType="Tag" entityID="00:01:02:03:04:06"/>
```

Location Change

An example:

```
<MovementTrackEvent distance="158.11388300841898" triggerDistance="5.0"
reference="marker1" referenceObjectID="1" trackDefn="defn1" entityType="Mobile Station"
entityID="00:01:02:03:04:05"/>
```

Chokepoint Condition

An example:

```
<ChokepointTrackEvent
lastSeen="11:10:08 PST 18 Jan 2007"
chokepointMac="00:0c:cc:60:13:a3"
chokepointName= "chokeA3"
trackDefn="choke"
entityType="Tag"
entityID="00:12:b8:00:20:4f"/>
```

Message format for the clear state.

An example:

```
<ChokepointTrackEvent
state="clear"
entityType="Tag"
trackDefn="choke"
entityID="00:12:b8:00:20:4f"/>
```

Emergency Condition

An example:

```
<ChokepointTrackEvent
lastSeen="11:36:46 PST Jan 18 2007"
emergencyReason= "detached"
trackDefn="emer"
entityType="Tag"
entityID="00:12:b8:00:20:50"/>
```

Note Emergency events are never cleared by location based services.

Notification Formats in Text

When you specify that notification be sent in Text format, the location server uses a plain-text string to indicate the condition. Following are examples:

```
Tag 00:02:02:03:03:04 is in Floor <floorName>
Tag 00:02:02:03:03:04 is outside Floor <floorName>
Client 00:02:02:03:09:09 is in Area <areaName>
RogueClient 00:02:02:08:08:08 is outside Building <buildingName>
Tag 00:02:02:03:03:06 has moved 105 feet where the trigger distance was 90 feet.
Tag 00:02:02:03:03:20 missing for 14 mins, last seen <timestamp>.
```

Note Cisco maintains the right to modify the Text notification Format, without notice, at any time.

Note XML is the recommended format for systems that need to parse or analyze notification contents.

Cisco WCS as a Notification Listener

Cisco WCS acts as a notification listener. WCS receives the notifications from location servers in the form of the trap `locationNotifyTrap` as part of the MIB file bsnwras.my. The location server stores the content of the notification message in XML format in the variable `locationNotifyContent` (see "Notification Formats in XML" section on page 6-11).

```
locationNotifyTrap  NOTIFICATION-TYPE
    OBJECTS { locationNotifyContent}
    STATUS current
    DESCRIPTION
        "This trap will be generated by the location server
        for notifications of location events."
    ::= { bsnTraps 89 }

locationNotifyContent OBJECT-TYPE
    SYNTAX  OCTET STRING(SIZE(0..512))
    MAX-ACCESS  accessible-for-notify
    STATUS  current
    DESCRIPTION
        "This is the content of the notification."
    ::= { bsnTrapVariable 72 }
```

WCS translates the traps into UI alerts and displays them in the following formats:

- Missing (Absence)

  ```
  Absence of Tag with MAC 00:0c:cc:5b:e4:1b, last seen at 16:19:45 13 Oct 2005.
  ```

- In/Out (Containment)

  ```
  Tag with MAC 00:0c:cc:5b:fa:44 is In the Area 'Rochester > Rochester > 5th Floor >
  nycTestArea'
  ```

- Distance

  ```
  Tag with MAC 00:0c:cc:5b:fa:47 has moved beyond the distance configured for the marker
  'marker2'.
  Tag with MAC 00:0c:cc:5b:f9:b9 has moved beyond 46.0 ft. of marker 'marker2', located
  at a range of 136.74526528595058 ft.
  ```

- Battery Level

  ```
  Tag 00:01:02:03:04:06 has medium battery, last seen 11:06:01 23 May 2006
  ```

- Location Change

  ```
  Mobile Station 00:01:02:03:04:05 has moved
  158.11388300841898ft, where the trigger distance was 5.0
  ```