

Zone Manager – IBM WebSphere Sensor Events Interface Specification

VERSION 2.4.0
AUGUST 23, 2011

RF Code, the RF Code logo, Mantis and Spider are registered trademarks of RF Code, Inc. All other product or service names are the property of their respective owners.

Information in this document is provided solely to enable system and software implementers to use RF Code products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document. RF Code reserves the right to make changes without further notice to any products described herein. RF Code makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does RF Code assume any liability arising out of the application or use of any product, and specifically disclaims any and all liability, including without limitation consequential or incidental damages.

Copyright Statement

Copyright © 2008-2011 RF Code, Inc. All rights reserved.

This document, as well as the hardware and firmware described therein, is furnished under license and may only be used or copied in accordance with the terms of such license. The information in these pages are furnished for informational use only, are subject to change without notice, and should not be construed as a commitment by RF Code, Inc. RF Code assumes no responsibility or liability for any errors or inaccuracies that may appear in these pages.
Every effort has been made to supply complete and accurate information. However, RF Code assumes no responsibility for its use, or for any infringements of patents or other rights of third parties, which would result. Information in these pages is subject to change without notice.

RF Code, Inc.
9229 Waterford Centre Blvd
Suite 500
Austin, TX 78758
www.rfcode.com

1 Introduction

This specification defines and describes the interface with IBM WebSphere Sensor Events interface, as implemented in RF Code Zone Manager V2.3.0 and later. This interface allows tag and reader status information to be delivered, real-time, to an IBM WebSphere Premises V6.1 server or an IBM WebSphere Sensor Events V7.x server, and applications associated with that server. As currently defined, the interface does not include mechanisms for control or configuration of the RF Code Zone Manager product by Premises or its applications, other than the existing Zone Manager API (as detailed in the "RF Code Zone Manager API Specification" document). The Sensor Events interface, defined by IBM, has two distinct versions that are not compatible with one another. Consequently the event mapping produced by Zone Manager for these version is unique to each of them.

Any questions, comments, or suggestions concerning this specification can be sent to the RF Code Developer Support e-mail address, sdk@rfcode.com.

1.2 General System Structure

The RF Code Zone Manager server ("Zone Manager") consists of a single application server, supported on Red Hat Linux systems and Microsoft Windows XP and Vista systems (both 32-bit and 64-bit Intel architecture). The interfaces provided by product are consistent between all supported operating systems.

IBM Sensor Events servers are complex, enterprise class server applications, supporting a variety of server topologies. The Sensor Events interfaces provided by IBM WebSphere Premises v6.x and IBM WebSphere Sensor Events v7.x support, among other things, HTTP-based interfaces allowing the delivery of messages to the server system.

In the v6.x interface, the messages are encoded to conform to the "IBMSensorEvent" schema. Each message can include one sensor event, and each event can include a variety of attribute-value style data. Delivery to the Sensor Events server of the XML-encoded messages is done by issuing an HTTP POST to the http://<hostname>:<port>/ibmse/eventpublish URL, with the event XML passed as the value of the "eventxml" parameter (in standard "url-encoded" format).

In the v7.x interface, the messages are encoded to conform to a new, and incompatible, version of the "IBMSensorEvent" schema. Each message can include one or more sensor events of the same type, and each event can include a variety of pre-defined attributes, as well as generic attribute-value pairs (for data not covered by the predefined schema). A number of well-known event types are defined, along with specific expected behaviors associated with those events. Delivery to the Sensor Events server of the XML-encoded messages is done by issuing an HTTP POST to the http://<hostname>:<port>/ibmse/gateway/eventpublish URL, with the event XML passed as the value of the "eventxml" parameter (in standard "url-encoded" format).

1.3 Zone Manager Configuration

Enabling and configuring the support in Zone Manager for delivering events to the IBM Sensor Events v6.x interface, as implemented on IBM WebSphere Premises v6.x, includes three key steps:

1. Copying the ibmse_event_model*.jar files from Premises (typically found in C:\Program Files\IBM\RFID\premises\api\lib) to the jetty/libs directory under the install directory for Zone Manager (typically c:\Program Files\RF Code\zonemgr\jetty\libs or /usr/share/rfcode/zonemgr/jetty/libs).
2. Restart the Zone Manager server
3. Use the 'configset' command on Zone Manager to set the host name, port, and source ID to use for the events sent to the Premises server.

For the v7.x interface, only the third step (using 'configset' to set the options), with the addition of the requirement to set 'premises_version' to '7' ('6', corresponding to the v6.x API, is the default).

The configuration settings used by Zone Manager for the IBM Sensor Events integration module include the following:

- 'premises_hostname' – the host name or IP address of the Sensor Events server. Must be defined in order for the interface to be active.
- 'premises_port' – the port number of the HTTP interface on the Sensor Events server. If not defined, the default value is 9080.
- 'premises_source' – the source ID to be used for messages : identifying the sender to Sensor Events server. If not defined, the default value is 'RFCodeZoneMgr'.
- 'premises_enabled' – controls whether the interface is enabled ('true') or disabled ('false'). If not defined, the value defaults to 'true' (enabled).
- 'premises_version' – controls the version of the Sensor Events API used. Currently supported values are '6' (default – for v6.x servers), and '7' (for v7.x servers).
- 'wse_zone_for_source' – if defined and set to 'true', this causes the 'source_id' for 'sensor_event' to be based on the location of the tag, versus the 'premises_source' setting (for v7.x only)
- 'wse_tag_notification' – if defined and set to 'true', this causes tag events to be mapped under the 'report/asset/observation/notification/*' type (for v7.x only)
- 'wse_max_loc_interval' – if defined and set to a value N above 0, this causes tag location events to be reported (with a minimum refresh period of N seconds, even when no update has occurred) (for v7.x only). Otherwise, 'zoneentry' and 'zoneexit' events are sent, and only when location changes have occurred.

Setting of the configuration is done using the 'configset' command, either via the Zone Manager telnet-like interface, or via its HTTP interface. Current configuration settings can be read using the 'configlist' command. Configuration changes are immediately applied (so, changing premises_hostname will immediately redirect messages to the server at the new host name).

1.4 Sensor Events v6.x API

1.4.1 Message Definitions

Once active, the Sensor Events integration module in Zone Manager enables hooks into the tag and reader status update streams within the Zone Manager server (the same update streams used to support the 'tagupdates' URL defined in the “RF Code Zone Manager API Specification'”). The module processes events associated with newly found tags, tag updates, and tag losses. Also, reader creation, update, and deletion events are processed. Unlike the 'tagupdates' interface, the integration module captures and reports the full set of attributes for any updated tag or reader (versus just the changed attributes), which make the reported messages more 'complete' or 'standalone', which may simplify processing by upstream applications in the Sensor Events server.

The messages sent are encoded per the IBMSensorEvent v6.1 schema, and include both header fields and meta-data attributes. The messages do not include a custom payload, avoiding the need for an installed custom payload class on the Sensor Events server.

The header fields are populated as follows:

- The EventType is set based on the type of message being reported. These types include:
  - 'RFCode/Tag/Found' – message is reporting a newly-observed tag and its attributes
  - 'RFCode/Tag/Update' – message is reporting the update of one or more attributes of a previously observed tag.
  - 'RFCode/Tag/Lost' – message is reporting the loss of contact with a previously observed tag.
  - 'RFCode/Reader/Create' – message is reporting a newly created reader, and its attributes and status.
  - 'RFCode/Reader/Update' – message is reporting a status or configuration change in an existing reader.
  - 'RFCode/Reader/Delete' – message is reporting the deletion of an existing reader.
- The 'SourceId' is set based on the 'premises_source' configuration setting (default is 'RFCodeZoneMgr').

- If the messages is a tag message, and the tag has a value for its 'location' attribute (due to matching a location rule, the 'GeoLocation' attribute is set to the hierarchical representation of the location attribute (i.e. /usa/texas/austin, if the location were 'austin', which had a parent location of 'texas' and a grandparent of 'usa').

The message meta-data is more variable, depending upon the specific type of tag or reader being reported.

For tags, the meta-data includes the following:

- A 'tagtype' attribute (a string) representing the type of tag (i.e. 'mantis04A').
- A 'tagid' attribute (a string) representing the globally unique ID of the tag
- A 'taggroup' attribute (a string) representing the ID of the tag group for the tag
- Direct mapping of all the tag's attributes ('motion', 'tamper', 'lowbattery', 'panic', etc), using the matching data type (boolean, integer, etc).
- A 'location' attribute (a string) representing the ID of the tag's matching location (if any)
- All attributes inherited from the tag's matching location. These attributes are strings, and will have IDs consisting of the string 'location.' followed by their ID in the location object (i.e. The location attribute 'city' would be mapped to 'location.city' in the event).

For readers, the meta-data includes the following:

- A 'readertype' attribute (a string) corresponding to the type of reader (i.e. 'M200')
- A 'readerid' attribute (a string) corresponding to the ID of the reader
- An 'enabled' attribute (a boolean) corresponding to the enable/disable state of the reader
- A 'state' attribute (a string) corresponding to the current operating status of the reader
- A 'groups' attribute (a string list) corresponding to the tag group IDs configured for the reader
- A direct mapping of all the reader-specific setting attributes (except the 'password' attribute, if defined), using the corresponding data type (boolean, integer, string, etc.)

1.4.2 Interface Behaviors

The interface module attempts to maintain a robust event interface while avoiding the problem of unbounded growth of the queue of events during a prolonged outage (as defined by the inability to successfully deliver events to the Sensor Events server).

Whenever an attempt to deliver an event to the Sensor Events server fails, the event remains on the queue and is retried every 5 seconds until successful (or until Zone Manager is restarted or the Sensor Events interface is disabled). Attempts to connect to the Sensor Events server will time-out if the server takes longer than 30 seconds to accept the connection.

To avoid unbounded growth, only one 'Update' message from a given tag or a given reader is allowed in the pending event queue at a given time. If a new update is reported, its content replaces the pending update message in the event queue. Under normal conditions, events are delivered quickly enough to avoid having new updates replace pending ones.

1.5 Sensor Events v7.x API

1.5.1 Message Definitions

Once active, the Sensor Events integration module in Zone Manager enables hooks into the tag and reader status update streams within the Zone Manager server (the same update streams used to support the 'tagupdates' URL defined in the "RF Code Zone Manager API Specification'"). The module processes events associated with newly found tags, tag updates, and tag losses. Also, reader creation, update, and deletion events are processed. Unlike the 'tagupdates' interface, the integration module captures and reports the full set of attributes for any updated tag or reader (versus just the changed attributes), which make the reported messages more 'complete' or 'standalone', which may simplify processing by upstream applications in the Sensor Events server.

The messages sent are encoded per the IBMSensorEvent v7.0 schema, and include both header fields and meta-data attributes.

The header fields are populated as follows:

- The event_type is set based on the type of message being reported. These types include:
  - 'report/asset/observation/tagfound' – message is reporting a newly-observed tag and its attributes (Note: if 'wse_tag_notification' is set to true, 'report/asset/observation/notification/tagfound' events are sent instead)
  - 'report/asset/observation/tagupdate' – message is reporting the update of one or more attributes of a previously observed tag. (Note: if 'wse_tag_notification' is set to true, 'report/asset/observation/notification/tagupdate' events are sent instead)
  - 'report/asset/observation/taglost' – message is reporting the loss of contact with a previously observed tag. (Note: if 'wse_tag_notification' is set to true, 'report/asset/observation/notification/taglost' events are sent instead)
  - 'report/diagnostic/reader/create' – message is reporting a newly created reader, and its attributes and status.
  - 'report/diagnostic/reader/update' – message is reporting a status or configuration change in an existing reader.
  - 'report/diagnostic/reader/delete' – message is reporting the deletion of an existing reader.
  - 'report/asset/observation/location/zoneentry' – message sent when a tag enters a new location zone (only sent if 'wse_max_loc_interval' is undefined or set to 0)
  - 'report/asset/observation/location/zoneexit' – message sent when a tag exits an existing location zone (only sent if 'wse_max_loc_interval' is undefined or set to 0)
  - 'report/asset/observation/location' – message sent when a tag changes location, or 'wse_max_loc_interval' seconds has elapsed since the last location event (this is only sent if 'wse_max_loc_interval' is greater than 0).
  - 'report/diagnostic/alert' – message sent when a low battery condition exists for a tag (only sent if 'wse_tag_notification' is set to true).

- The 'source_id' is set based on the 'premises_source' configuration setting (default is 'RFCodeZoneMgr'). If 'wse_zone_for_source', the 'source_id' for the 'sensor_event' is instead set to the location path (as reported for the 'zone' attribute, below) – the 'source_id' for the 'sensor_events' is still based on the 'premises_source' configuration setting.

- If the messages is a tag message, and the tag has a value for its 'location' attribute (due to matching a location rule), the 'zone' attribute under the 'location' section will be set to the hierarchical representation of the location attribute (i.e. /usa/texas/austin, if the location were 'austin', which had a parent location of 'texas' and a grandparent of 'usa'). An unknown location is reported as '/'.
- The following tag attributes are reported using well-known tags under the 'condition' section of the corresponding 'sensor_event':
  - 'lowbattery' – reported using 'battery_status', and reporting a value of '0' (low) or '5' (normal).
  - 'panic' – reported using 'buttons_pressed' with a value of 'Panic' if true (blank otherwise).

- 'temp' – reported using 'temperature' tag, with the value in Celsius ('temperature_units' are set to 'Celsius').
- 'humidity' – reported using 'humidity', with the value in percent.

- All other tag attributes (of data type Boolean, integer, float, or string) are reported using the 'field' tag.
- The tag group is reported using the 'group' tag under the 'subject' tag, while the tag's unique ID is reported using the 'id' tag.

- Reader attributes are mapped using the 'field' tag.

1.5.2 Updated Behaviors for v7.x for IBM RTLS (new for 2.4.0)

As of Zone Manager v2.4.0, several updated behaviors have been added to support additional requirements by IBM RTLS implementations. By default, the previous behaviors are maintained, and activating these new behaviors requires the setting of additional configuration options, using the 'configset' command.

First, the 'wse_zone_for_source' setting can be set to 'true', which will alter the behavior of the <source_id> attribute reported for the <sensor_event> instances. If set, the value in this field will be be location path (as currently shown in the 'location' attribute, described above), instead of the value of the 'premises_source' setting (as done by default).

Second, the 'wse_tag_notification' setting can be set to 'true' which will alter the event types used for reporting tag events. Specifically, each of the 'report/asset/observation/tag*' events documented previously will be reported as 'report/asset/observation/notification/tag*' (e.g. report/asset/observation/tagfound is instead reported as report/asset/observation/notification/tagfound). Also, low battery conditions will result in an additional event, of type 'report/diagnostic/alert'.

Third, the 'wse_max_loc_interval' setting can be set to a numeric value representing a maximum period of time (in seconds) between reports of a tag's location. If set a value N that is greater than 0, this will cause 'report/asset/observation/location/zoneentry' and 'report/asset/observation/location/zoneexit' events to instead be reported as 'report/asset/observation/location' events, and for these events to be repeated every N seconds (even when no location change has occurred. If used, N should be set as high as can be allowed by the application, as repeated reporting of unchanged tag locations can generate significant traffic. If 'wse_max_loc_interval' is undefined, or set to 0, 'zoneentry' and 'zoneexit' events are reported, as previously described.

1.5.3 Interface Behaviors

The interface module attempts to maintain a robust event interface while avoiding the problem of unbounded growth of the queue of events during a prolonged outage (as defined by the inability to successfully deliver events to the Sensor Events server).

Whenever an attempt to deliver an event to the Sensor Events server fails, the event remains on the queue and is retried every 5 seconds until successful (or until Zone Manager is restarted or the Sensor Events interface is disabled). Attempts to connect to the Sensor Events server will time-out if the server takes longer than 30 seconds to accept the connection.

To avoid unbounded growth, only one 'Update' message from a given tag or a given reader is allowed in the pending event queue at a given time. If a new update is reported, its content replaces the pending update message in the event queue. Under normal conditions, events are delivered quickly enough to avoid having new updates replace pending ones.