rainforest automation

RAVEn™

Radio Adaptor for Viewing Energy

XML API Manual

Version 1.27
March 2013

RAVEn™ – Radio Adapter for Viewing Energy

RFA-Z106
Version 1.27

XML API Manual

Table of Contents

OVERVIEW

RAVEn™ XML API

The **RAVEn**™ is a USB device that communicates with a smart meter over a secured ZigBee wireless network. The **RAVEn**™ is an endpoint on the network that is authorized and authenticated to communicate with the smart meter. The XML (eXtensible Markup Language) API (Application Programming Interface) described in this document provides a mechanism to allow external applications to receive smart meter data from the **RAVEn**™.

Communications Model

The **RAVEn**™ is a USB device; the USB port is mapped to a virtual serial port, which shows up as a standard COM port. Applications that can connect to a COM port can also connect to the **RAVEn**™. The **RAVEn**™ sends and receives data as a serial stream. The **RAVEn**™ uses an 8 bit Extended ASCII character set (code page 1252) to encode the data. This may change in the future in order to support UTF-8.

Settings for the virtual serial COM port:

- `PORT NAME: determined by OS`
- `BAUD RATE: 115,200`
- `DATA BITS: 8`
- `PARITY: N`
- `STOP BITS: 1`

The **RAVEn**™ uses an asynchronous serial communications model. This means that the transmit and receive data streams are independent; the **RAVEn**™ can be sending and receiving data at the same time. The **RAVEn**™ does not use hand shaking, acknowledgements, or time synchronization.

The **RAVEn**™ is also a ZigBee device. It is ZigBee Smart Energy 1.0 compliant, and is certified by the ZigBee Alliance to operate according to that standard. Therefore, the XML API options are restricted to what is allowed by the ZigBee Smart Energy 1.0 standard. The **RAVEn**™ must be authorized and authenticated by the ZigBee Coordinator before the **RAVEn**™ can communicate with the smart meter. Generally, the smart meter is also the ZigBee Coordinator. The **RAVEn**™ is authorized by the owner of the smart meter (i.e. the electric utility); the owner needs the MAC ID and Install Code for the **RAVEn**™ in order to set up the authorization.

Data Structures

The **RAVEn**™ sends and receives data as a serial stream. The data is structured as XML Fragments. An XML Fragment is a stripped down XML Element. The **RAVEn**™ uses XML Fragments to simplify the parsing of the data stream, while providing a data structure that is flexible and human readable.

The rules for XML Fragments are:

1. XML Fragments are transmitted as a stream; there are no BOF or EOF markers.
2. An XML Fragment is a well formed XML Element.
3. XML Elements do not have XML Attributes.
4. Only the root XML Element has child XML Elements.
5. There is no XML Declaration Section.
6. There are no XML Name spaces.

The **RAVEn**™ receives commands and sends notifications. A command is a request to the **RAVEn**™ to do something. The **RAVEn**™ will execute the command, which will trigger an event. **RAVEn**™ events send out notifications that contain information about the event.

For example:

1. Send a GET_CURRENT_PRICE command, which causes the **RAVEn**™ to request the current price stored in the meter;
2. **RAVEn**™ returns a *PriceCluster* notification with the current price received from the meter.

The **RAVEn**™ will also send notifications when events are triggered by something other than a command. For instance, when the **RAVEn**™ receives a new text message from the smart meter, the **RAVEn**™ will send a *MessageCluster* notification. This is why the communications model is asynchronous; the **RAVEn**™ sends notifications whenever an event occurs, in addition to when a command is received.

The format of an XML Fragment is

```
<tag>
      [<string>text</string>]
      [<hex>0xFFEEDDCCBBAA99887766554433221100</hex>]
      [<int>-987654321</int>]
      [<decimal>-7654321.12345</decimal>]
      [<![CDATA[binary]]>]
      [<enumeration>{A|B|C}</enumeration>]
</tag>
```

Where:

- `<tag>` is the tag for the root XML Element
- `<string>` is the start tag for an element with Extended ASCII string
- `<hex>` is the start tag for an element with a hexadecimal data
- `<int>` is the start tag for an element with a signed 32bit integer

- `<decimal>` is the start tag for an element with a signed decimal(12.5)
- `<base64>` is the start tag for an element with Base64 data
- `<![CDATA[binary]]>` is the tag for a CDATA section
- `binary` is the binary content of the `CDATA` section
- `[ ]` brackets indicate optional elements
- `<enumeration>` is the start tag for an element that can have a specific list of values.
- `{A|B|C}` are the different options for the element value
- `[<Element> | <Element>]` are the different optional elements
- Element names are case insensitive; the case is used strictly for legibility

RAVEN™ receives commands with the following XML Fragment structure:

```
<Command>
      <Name>text</Name>
      …
</Command>
```

Where:

- `<Command>` is the start tag for the command XML Fragment
- `<Name>` is the start tag for the name of the command
- … indicates the variable number of command specific parameters

RAVEN™ sends notifications with the following XML Fragment structure:

```
<tag>
      <element>value</element>
      …
</tag>
```

Where:

- `<tag>` is the start tag for the XML Fragment; each notification will have a unique tag name;
- `<element>` is the start tag for an element; there will be one or more child elements in the fragment; each notification element will have a unique element name.
- … indicates the variable number of notification specific elements

The first element of every notification is always the MAC ID of the **RAVEn**™ that generated the notification.

Note: the **RAVEn**™ does not send or require the MAC ID of the meter when in single meter operation. This may change in the future. For multiple meters, the meter MAC ID will always be required.

Features

RAVEN™ XML Fragments are loosely organized as Features, where each Feature is a logical grouping of notifications and commands.

The API is organized into these Features:

Feature	Description
RAVEn™	**RAVEn™** device specific information
Meter	Meter specific information
Time	Time and Date information
Message	Text Message information
Price	Price information
Simple Metering	Metering data information
Firmware Update	Firmware upgrade procedure

FEATURES

RAVEn™ Feature

The **RAVEn™** Feature provides the commands and notifications for managing the configuration of the **RAVEn™** and diagnosing the communications to the smart meter.

Key commands and notifications include:

- Start up, joining, and connectivity status notifications
- Restarting the **RAVEn™** (Soft Reboot)
- Resetting the **RAVEn™** (Factory Reset, will Decommission device)
- Get revision information for **RAVEn™**
- Configuring the scheduler on the **RAVEn™**

1. Command: INITIALIZE

Send the INITIALIZE command to have the **RAVEn™** reinitialize the XML parser. Use this command when first connecting to the **RAVEn™** prior to sending any other commands. While initialization is not required, it will speed up the initial connection.

```
<Command>
      <Name>initialize</Name>
</Command>
```

2. Command: RESTART

Send the RESTART command to have the **RAVEn™** go through the start-up sequence. This command is useful for capturing any diagnostic information sent during the start-up sequence.

```
<Command>
      <Name>restart</Name>
</Command>
```

3. Command: FACTORY_RESET

Send the FACTORY_RESET command to decommission the **RAVEn™**. This command erases the commissioning data and forces a restart. On restart, the **RAVEn™** will begin the commissioning cycle.

```
<Command>
      <Name>factory_reset</Name>
</Command>
```

4. Command: GET_CONNECTION_STATUS

Send the GET_CONNECTION_STATUS command to get the **RAVEn™** connection information. The **RAVEn™** will send a *ConnectionStatus* notification in response.

```
<Command>
      <Name>get_connection_status</Name>
</Command>
```

5. Notify: *ConnectionStatus*

The **RAVEn™** will send notifications during the start-up sequence and during the join/re-join sequence. These notifications are useful for diagnostic purposes.

```
<ConnectionStatus>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <Status>{enumeration}</Status>
      [<Description>{string}</Description>]
      [<StatusCode>0xFF</StatusCode>]
      [<ExtPanId>0xFFFFFFFFFFFFFFFF</ExtPanId>]
      [<Channel>00</Channel>]
      [<ShortAddr>0xFFFF</ShortAddr>]
      <LinkStrength>0xFF</LinkStrength>
</ConnectionStatus>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
Status	Initializing | Network Discovery | Joining | Join: Fail | Join: Success | Authenticating | Authenticating: Success | Authenticating: Fail | Connected | Disconnected | Rejoining	Indicates the current state of the device
Description	Text; Optional	Gives a description of the device state
StatusCode	0x00 – 0xFF; Optional	If available, provides a status code for the current state
ExtPanId	0x0 - 0xFFFFFFFFFFFFFFFF; Optional	Provides the extended PAN ID of the network the device is trying to join or has joined
Channel	11 – 26; Optional	Indicates the channel on which the device is operating
ShortAddr	0x0000 – 0xFFFF; Optional	The short address assigned to the **RAVEn™** by the network coordinator
LinkStrength	0x00 – 0x64	Indicates the strength of the link

6. Command: GET_DEVICE_INFO

Send the GET_DEVICE_INFO command to get the **RAVEn™** configuration information. The **RAVEn™** will send a DeviceInfo notification in response.

```
<Command>
      <Name>get_device_info</Name>
</Command>
```

7. Notify: *DeviceInfo*

DeviceInfo notifications provide some basic information about the **RAVEn™**.

```
<DeviceInfo>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <InstallCode>0xFFFFFFFFFFFFFFFF</InstallCode>
      <LinkKey>0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF</LinkKey>
      <FWVersion>{string}</FWVersion>
      <HWVersion>{string}</HWVersion>
      <ImageType>{string}</ImageType>
      <Manufacturer>{string}</Manufacturer>
      <ModelId>{string}</ModelId>
      <DateCode>{string}</DateCode>
</DeviceInfo>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
InstallCode	0xFFFFFFFFFFFFFFFF	Install Code
LinkKey	0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	Link Key
FWVersion	Text	Firmware Version
HWVersion	Text	Hardware Version
ImageType	Text	Firmware Image Detail
Manufacturer	Text	“Rainforest Automation”
ModelId	Text	“RFA-Z106”
DateCode	YYYYMMDDZZZZZZZZ	Manufacturer’s date code and lot number

8. Command: GET_SCHEDULE

Send the GET_SCHEDULE command to get the **RAVEn™** scheduler information. The **RAVEn™** will send the *ScheduleInfo* notification in response; or, **RAVEn™** will send a series of *ScheduleInfo* notifications if the Event field is omitted.

```
<Command>
      <Name>get_schedule</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<Event>{enumeration}</Event>]
</Command>
```

Element	Range	Description

MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Event	time | price | demand | summation | message	The type of event being scheduled

9. Notify: *ScheduleInfo*

ScheduleInfo notifications provide the frequency at which a certain event is read and if it is at present enabled or disabled.

```
<ScheduleInfo>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      <Event>{enumeration}</Event>
      <Frequency>0x00000000</Frequency>
      <Enabled>{enumeration}</Enabled>
</ScheduleInfo>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Event	time | price | demand | summation | message	The type of event being scheduled
Frequency	0x0 - 0xFFFFFFFE	The frequency in seconds the event will be executed
Enabled	Y | N	Y: the scheduled event will execute; N: the scheduled event will not execute.

10. Command: SET_SCHEDULE

Send the SET_SCHEDULE command to update the **RAVEn™** scheduler. The command options include setting the frequency of the command in seconds, and disabling the event. If the event is disabled the frequency is set to 0xFFFFFFFF.

```
<Command>
      <Name>set_schedule</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      <Event>{enumeration}</Event>
      <Frequency>0x00000000</Frequency>
      <Enabled>{enumeration}</Enabled>
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Event	time | price | demand | summation | message	The type of event being scheduled
Frequency	0x0 - 0xFFFFFFFE	The frequency in seconds the event will be executed

Enabled	Y | N	Y: the scheduled event will execute (default); N: the scheduled event will not execute.

11. Command: SET_SCHEDULE_DEFAULT

Send the SET_SCHEDULE_DEFAULT command to reset the **RAVEn™** scheduler to default settings. If the Event field is set, only that schedule item is reset to default values; otherwise all schedule items are reset to their default values.

```
<Command>
      <Name>set_schedule_default</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<Event>{enumeration}</Event>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Event	time | price | demand | summation | message	The type of event being scheduled; if omitted, command applies to all scheduled items.

12. Command: GET_METER_LIST

Send the GET_METER_LIST command to get the list of meters the **RAVEn™** is connected to. The **RAVEn™** will send a *MeterList* notification in response.

```
<Command>
      <Name>get_meter_list</Name>
</Command>
```

13. Notify: *MeterList*

MeterList notifications provide a list of meters the **RAVEn™** is connected to.

```
<MeterList>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [ … ]
</MeterList>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional	MAC Address of Meter; there can any number of MeterMacId tags in the *MeterList* fragment

Meter Feature

The Meter Feature provides the commands and notifications for getting information about the meter and the network the device is on.

Key commands and notifications include:

- Getting information about the type of meter and its identifiers
- Finding the status of the network connection and the quality of the link

1. Command: GET_METER_INFO

Send the GET_METER_INFO Command to get the meter information. The **RAVEn™** will send a *MeterInfo* notification in response.

```
<Command>
      <Name>get_meter_info</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter

2. Notify: *MeterInfo*

MeterInfo notifications provide information about meters that are on the network.

```
<MeterInfo>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <MeterType>{enumeration}</MeterType>
      <NickName>{string}</NickName>
      [<Account>{string}</Account>]
      [<Auth>{string}</Auth>]
      [<Host>{string}</Host>]
      [<Enabled>{enumeration}</Enabled>]
</MeterInfo>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
Meter Type	electric | gas | water | other	Type of meter
Nickname	Text	Nickname set for the meter
Account	Text; Optional	Account Identification
Auth	Text; Optional	Authentication code
Host	Text; Optional	Hosting Provider

Enabled	Y | N; Optional	Y: to start transmitting data to host N: to stop transmitting data to host

3. Command: GET_NETWORK_INFO

Send the GET_NETWORK_STATUS Command to get the status of device on the network. The **RAVEn™** will send a *NetworkInfo* notification in response.

```
<Command>
      <Name>get_network_info</Name>
</Command>
```

4. Notify: *NetworkInfo*

NetworkInfo notifications provide information about the network that the device is on.

```
<NetworkInfo>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <CoordMacId>0xFFFFFFFFFFFFFFFF</CoordMacId>
      <Status>{enumeration}</Status>
      <Description>{string}</Description>
      <StatusCode>[0xFF]</StatusCode>
      <ExtPanId>0xFFFFFFFFFFFFFFFF</ExtPanId>
      <Channel>00</Channel>
      <ShortAddr>0xFFFF</ShortAddr>
      <LinkStrength>0xFF</LinkStrength>
</NetworkInfo>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
CoordMacId	0xFFFFFFFFFFFFFFFF	MAC Address of the network coordinator (normally the meter)
Status	Initializing | Network Discovery | Joining | Join: Fail | Join: Success | Authenticating | Authenticating: Success | Authenticating: Fail| Connected| Disconnected| Rejoining	Indicates the current state of the device
Description	Text	Gives a description of the device state
StatusCode	0x00 – 0xFF	If available, provides a status code for the current state; if not available, then null
ExtPanId	0x0 - 0xFFFFFFFFFFFFFFFF	Provides the extended PAN ID of the network the device is trying to join or has joined
Channel	11 – 26	Indicates the channel on which the device is operating
ShortAddr	0x0000 – 0xFFFF	The short address assigned to the **RAVEn™**

		by the network coordinator
LinkStrength	0x00 – 0x64	Indicates the strength of the link

5. Command: SET_METER_INFO

Send the SET_METER_INFO Command to set the meter information.

```
<Command>
      <Name>set_meter_info</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<NickName>{string}</NickName>]
      [<Account>{string}</Account>]
      [<Auth>{string}</Auth>]
      [<host>{string}</host>]
      [<enabled>{enumeration}</enabled>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
NickName	Text; Optional	Meter nick name
Account	Text; Optional	Account Identification
Auth	Text; Optional	Authentication code
Host	Text; Optional	Hosting Provider
Enabled	Y | N; Optional	Y: to start transmitting data to host N: to stop transmitting data to host

Time Feature

The Time Feature provides the commands and notifications for getting the time from the smart meter.

Key commands and notifications include:

- Notify what the time is on a smart meter
- Ask for the latest time from a smart meter

1. Command: GET_TIME

Send the GET_TIME command to get the current time. The **RAVEn™** will send a *TimeCluster* notification in response.

```
<Command>
      <Name>get_time</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<Refresh>{enumeration}</Refresh>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Refresh	Y | N; Optional	Y: Get current time from meter N: Get time from **RAVEn™** cache (default)

2. Notify: *TimeCluster*

TimeCluster notifications provide the current time reported on the meter in both UTC and Local time. The time values are the number of seconds since 1-Jan-2000 UTC.

```
<TimeCluster>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <UTCTime>0xFFFFFFFF</UTCTime>
      <LocalTime>0xFFFFFFFF</LocalTime>
</TimeCluster>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
UTCTime	0x0 – 0xFFFFFFFF	UTC Time as reported by meter
LocalTime	0x0 – 0xFFFFFFFF	Local Time as reported by meter

Message Feature

The Message Feature provides the commands and notifications for managing messages routed through the smart meters.

Key commands and notifications include:

- Message notifications when a smart meter receives a message
- Message confirmation by the User

1. Command: GET_MESSAGE

Send the GET_MESSAGE command to have the **RAVEn™** get the current text message. The **RAVEn™** will send a *MessageCluster* notification in response.

```
<Command>
      <Name>get_message</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<Refresh>{text}</Refresh>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Refresh	Y | N; Optional	Y: Get current message from meter N: Get message from **RAVEn™** cache (default)

2. Notify: *MessageCluster*

MessageCluster notifications provide the current text message from the meter. If a confirmation is required, the ConfirmationRequired flag is set. If the user has already confirmed the message, then the Confirmed flag is set to Y. The ID is the reference to a particular message. The message text is HTML escape encoded.

```
<MessageCluster>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <TimeStamp>0xFFFFFFFF</TimeStamp>
      <Id>0xFFFFFFFF</Id>
      <Text>{string}</Text>
      <ConfirmationRequired>{enumeration}</ConfirmationRequired>
      <Confirmed>{enumeration}</Confirmed>
      <Queue>{enumeration}</Queue>
</MessageCluster>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
TimeStamp	0xFFFFFFFF	UTC Time when message cluster data was received from meter
Id	0x0 – 0xFFFFFFFF	Message ID from meter
Text	Text	Contents of message HTML encoded: > replaces the > character < replaces the < character & replaces the & character " replaces the " character
Confirmation Required	Y | N	Y: a user confirmation is required; N: a user confirmation is not required (default)
Confirmed	Y | N	Y: the user confirmation has been sent; N: the user confirmation has not been sent (default)
Queue	Active | Cancel Pending	Active: Indicates message is in active queue Cancel Pending: Indicates message is in cancel pending queue

3. Command: CONFIRM_MESSAGE

Send the CONFIRM_MESSAGE command to have the **RAVEn™** confirm the message as indicated by the ID. To verify that the message confirmation was sent, use a GET_MESSAGE command with Refresh=Y. The resulting *MessageCluster* notification should show Confirmed=Y.

```
<Command>
      <Name>confirm_message</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      <Id>0xFFFFFFFF</Id>
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
ID	0x0 – 0xFFFFFFFF	Message ID to confirm

Price Feature

The Price Feature provides the commands and notifications for managing prices from the smart meters.

Key commands and notifications include:

- Price notifications from a smart meter
- Price Tier notifications from a smart meter
- Set a user-defined Price when the smart meter does not provide a price

1. Command: GET_CURRENT_PRICE

Send the GET_CURRENT_PRICE command to get the price information. Set the Refresh element to Y to force the **RAVEn™** to get the information from the meter, not from cache.
The **RAVEn™** will send a *PriceCluster* notification in response.

```
<Command>
      <Name>get_current_price</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<Refresh>{enumeration}</Refresh>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Refresh	Y | N; Optional	If user price is set, this is ignored; price is always from **RAVEn™** cache. If user price is not set: Y: Get current price from meter N: Get price from **RAVEn™** cache (default)

2. Command: SET_CURRENT_PRICE

Send the SET_CURRENT_PRICE command to set the user-defined price on the **RAVEn™**. The Price field is an integer; the Trailing Digits field indicates where the decimal place goes (i.e., the divisor). The user-defined price will override the meter price. Setting the user-defined price to zero will clear the user entered price in the **RAVEn™**, and the meter price will be used, if available.

```
<Command>
      <Name>set_current_price</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      <Price>0x000000</Price>
      <TrailingDigits>0x00</TrailingDigits>
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Price	0x0 – 0xFFFFFFFF	Price to be set; set to zero to clear the user defined price and meter price will be used, if available
TrailingDigits	0x00 – 0xFF	The number of implicit decimal places in the price.

3. Notify: *PriceCluster*

PriceCluster notification provides the current price in effect on the meter, or the user-defined price set on the **RAVEn**™. If the user-defined price is set, the meter price is ignored. If the user-defined price is not set and the meter price is not set, then the price returned is zero. Either the TierLabel or the RateLabel, or neither, may be provided; for now, consider these labels as substitutes. The label provided is a **RAVEn**™ firmware compile option that is set to match the configuration of the smart meter.

```
<PriceCluster>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <TimeStamp>0xFFFFFFFF</TimeStamp>
      <Price>0x00000000</Price>
      <Currency>0x0000</Currency>
      <TrailingDigits>0x00</TrailingDigits>
      <Tier>0x00</Tier>
      [<TierLabel>{string}</TierLabel> |
      <RateLabel>{string}</RateLabel>]
</PriceCluster>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn**™
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
TimeStamp	0xFFFFFFFF	UTC Time when price cluster data was received from meter or set by user
Price	0x0 – 0xFFFFFFFF	Price on meter; will be zero if no price is set
Currency	0x0000	Currency being used; value of this field matches the values defined by ISO 4217
TrailingDigits	0x00 – 0xFF	The number of implicit decimal places in the price. (e.g. 2 means divide price by 100).
Tier	0x00 – 0xFF	The price Tier in effect.
TierLabel	Text; Optional	Tier label for the current price tier; will be “Set by User” if a user-defined price is set
RateLabel	Text; Optional	Rate label for the current price; will be “Set by User” if a user-defined price is set

Simple Metering Feature

The Simple Metering Feature provides the commands and notifications for managing the smart meter readings.

Key commands and notifications include:

- Summation notifications from a smart meter
- Instantaneous notifications from a smart meter
- Current Period Summation notification from the **RAVEn™**
- Last Period Summation notification from the **RAVEn™**
- Fast Poll mode allows the **RAVEn™** to return almost real time data readings from the smart meter; usually for when the User wants to see the effect of turning something on or off.

Reading Calculations: readings are recorded as integers and are converted into decimal number by using the multiplier and divisor. If the multiplier or divisor is zero then use a value of one instead.

For example:

- Reading: 123456
- Multiplier: 2
- Divisor: 10000
- Result: 24.6912

Formatting Hints: formatting hints are the preferred display settings as set by the utility; the user can override these settings.

For example:

- Reading: 12.3456
- Digits Left: 3
- Digits Right: 5
- Suppress Leading Zeros: False
- Formats as: 012.34560

1. Command: GET_INSTANTANEOUS_DEMAND

Send the GET_INSTANTANEOUS_DEMAND command to get the demand information from the **RAVEn™**. Set the Refresh element to Y to force the **RAVEn™** to get the information from the meter, rather than its local cache. The **RAVEn™** will send an *InstantaneousDemand* notification in response.

```
<Command>
      <Name>get_instantaneous_demand</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<Refresh>{enumeration}</Refresh>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Refresh	Y | N; Optional	Y: Get current reading from meter N: Get data from **RAVEn™** cache (default)

2. Notify: *InstantaneousDemand*

InstantaneousDemand notification provides the current consumption rate as recorded by the meter.

```
<InstantaneousDemand>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <TimeStamp>0xFFFFFFFF</TimeStamp>
      <Demand>0x000000</Demand>
      <Multiplier>0x00000000</Multiplier>
      <Divisor>0x00000000</Divisor>
      <DigitsRight>0x00</DigitsRight>
      <DigitsLeft>0x00</DigitsLeft>
      <SuppressLeadingZero>{enumeration}</SuppressLeadingZero>
</InstantaneousDemand>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
TimeStamp	0xFFFFFFFF	UTC Time when demand data was received from meter
Demand	0x0 – 0xFFFFFF	The raw instantaneous demand
Multiplier	0x0 – 0xFFFFFFFF	The multiplier; ignore if zero
Divisor	0x0 – 0xFFFFFFFF	The divisor; ignore if zero
DigitsRight	0x00 – 0xFF	Number of digits to the right of the decimal point to display
DigitsLeft	0x00 – 0xFF	Number of digits to the left of the decimal point to display
Suppress LeadingZero	Y | N	Y: Do not display leading zeros N: Display leading zeros

3. Command: GET_CURRENT_SUMMATION_DELIVERED

Send the GET_CURRENT_SUMMATION_DELIVERED command to get the summation data from the **RAVEn™**. Set the Refresh element to Y to force the **RAVEn™** to get the data from the meter, rather than its local cache. The **RAVEn™** will send a *CurrentSummationDelivered* notification in response.

```
<Command>
      <Name>get_current_summation_delivered</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      [<Refresh>{enumeration}</Refresh>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Refresh	Y | N; Optional	Y: Get current reading from meter N: Get summation data from **RAVEn™** cache (default)

4. Notify: *CurrentSummationDelivered*

CurrentSummationDelivered notification provides the total consumption to date as recorded by the meter.

```
<CurrentSummationDelivered>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <TimeStamp>0xFFFFFFFF</TimeStamp>
      <SummationDelivered>0x00000000</SummationDelivered>
      <SummationReceived>0x00000000</SummationReceived>
      <Multiplier>0x00000000</Multiplier>
      <Divisor>0x00000000</Divisor>
      <DigitsRight>0x00</DigitsRight>
      <DigitsLeft>0x00</DigitsLeft>
      <SuppressLeadingZero>{enumeration}</SuppressLeadingZero>
</CurrentSummationDelivered>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
TimeStamp	0xFFFFFFFF	UTC Time when summation data was received from meter
Summation Delivered	0x0 – 0xFFFFFFFF	The raw meter reading of the total summation of commodity delivered from the utility to the user.
Summation Received	0x0 – 0xFFFFFFFF	Total summation of commodity received from the user by the utility.
Multiplier	0x0 – 0xFFFFFFFF	The multiplier; ignore if zero

Divisor	0x0 – 0xFFFFFFFF	The divisor; ignore if zero
DigitsRight	0x00 – 0xFF	Number of digits to the right of the decimal point to display
DigitsLeft	0x00 – 0xFF	Number of digits to the left of the decimal point to display
Suppress LeadingZero	Y | N	Y: Do not display leading zeros N: Display leading zeros

5. Command: GET_CURRENT_PERIOD_USAGE

Send the GET_CURRENT_PERIOD_USAGE command to get the accumulated usage information from the **RAVEn™**. The **RAVEn™** will send a *CurrentPeriodUsage* notification in response. Note that this command will not cause the current period consumption total to be updated. To do this, send a GET_CURRENT_SUMMATION_DELIVERED command with Refresh set to Y.

```
<Command>
      <Name>get_current_period_usage</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter

6. Notify: *CurrentPeriodUsage*

CurrentPeriodUsage notification provides the total consumption for the current accumulation period, as calculated by the **RAVEn™**. The Multiplier and Divisor are used to calculate the actual decimal value from the CurrentPeriod, which is an integer. If the Multiplier and Divisor are Zero, then ignore them for calculation purposes (i.e., treat them as a value of one). The DigitsRight and DigitsLeft are formatting hints for the data. These indicate what the recommended formatting is for the value. The SuppressLeadingZero flag overrides the DigitsLeft formatting hint. StartDate is a UTC timestamp indicating when the current period started.

```
<CurrentPeriodUsage>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <TimeStamp>0xFFFFFFFF</TimeStamp>
      <CurrentUsage>0x00000000</CurrentUsage>
      <Multiplier>0x00000000</Multiplier>
      <Divisor>0x00000000</Divisor>
      <DigitsRight>0x00</DigitsRight>
      <DigitsLeft>0x00</DigitsLeft>
      <SuppressLeadingZero>{enumeration}</SuppressLeadingZero>
      <StartDate>0xFFFFFFFF</StartDate>
</CurrentPeriodUsage>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
TimeStamp	0xFFFFFFFF	UTC Time when the current period consumption total was last updated
CurrentUsage	0x0 – 0xFFFFFFFF	The current period total consumption value
Multiplier	0x0 – 0xFFFFFFFF	The multiplier; ignore if zero
Divisor	0x0 – 0xFFFFFFFF	The divisor; ignore if zero
DigitsRight	0x00 – 0xFF	Number of digits to the right of the decimal point to display
DigitsLeft	0x00 – 0xFF	Number of digits to the left of the decimal point to display
Suppress LeadingZero	Y | N	Y: Do not display leading zeros N: Display leading zeros
StartDate	0x0 – 0xFFFFFFFF	UTC Time of the start of the current period data accumulation

7. Command: GET_LAST_PERIOD_USAGE

Send the GET_LAST_PERIOD_USAGE command to get the previous period accumulation data from the **RAVEn™**. The **RAVEn™** will send a *LastPeriodUsage* notification in response.

```
<Command>
      <Name>get_last_period_usage</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter

8. Notify: *LastPeriodUsage*

LastPeriodUsage notification provides the total consumption for the previous accumulation period as calculated by the **RAVEn™**. The Start Date and End Date are UTC timestamps indicating the start and end times that define the previous period.

```
<LastPeriodUsage>
      <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
      <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
      <LastUsage>0x00000000</LastUsage>
      <Multiplier>0x00000000</Multiplier>
      <Divisor>0x00000000</Divisor>
      <DigitsRight>0x00</DigitsRight>
      <DigitsLeft>0x00</DigitsLeft>
      <SuppressLeadingZero>{enumeration}</SuppressLeadingZero>
      <StartDate>0xFFFFFFFF</StartDate>
      <EndDate>0xFFFFFFFF</EndDate>
</LastPeriodUsage>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
LastUsage	0x0 – 0xFFFFFFFF	The previous period total consumption value
Multiplier	0x0 – 0xFFFFFFFF	The multiplier; ignore if zero
Divisor	0x0 – 0xFFFFFFFF	The divisor; ignore if zero
DigitsRight	0x00 – 0xFF	Number of digits to the right of the decimal point to display
DigitsLeft	0x00 – 0xFF	Number of digits to the left of the decimal point to display
Suppress Leading Zero	Y | N	Y: Do not display leading zeros N: Display leading zeros
StartDate	0x0 – 0xFFFFFFFF	UTC Time of the start of the previous period
EndDate	0x0 – 0xFFFFFFFF	UTC Time of the end of the previous period

9. Command: CLOSE_CURRENT_PERIOD

Send the CLOSE_CURRENT_PERIOD command to have the **RAVEn™** roll over the current period to the last period and to initialize the current period.

```
<Command>
      <Name>close_current_period</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter

10. Command: SET_FAST_POLL

Send the SET_FAST_POLL command to have the **RAVEn**™ set the fast poll mode on the meter. In fast poll mode, the meter will send Instantaneous Demand updates at the frequency requested. This is a ZigBee Smart Energy 1.1 feature.

For ZigBee Smart Energy 1.0 meters, the **RAVEn**™ will emulate this feature, if possible. For some meters fast poll mode will not be allowed. In that case, polling will default to a maximum frequency of every 4 seconds for up to 15 minutes.

```
<Command>
      <Name>set_fast_poll</Name>
      [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
      <Frequency>0x0000</Frequency>
      <Duration>0x0000</Duration>
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
Frequency	0x04 – 0xFFFF	Frequency to poll meter, in seconds
Duration	0x0 – 0x0384	Duration of fast poll mode, in seconds; maximum is 15 minutes

11. Command: GET_PROFILE_DATA

Send the GET_PROFILE_DATA command to get the **RAVEn**™ to retrieve the interval data information from the meter. The **RAVEn**™ will send a *ProfileData* notification in response.

```
<Command>
  <Name>get_profile_data</Name>
  [<MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>]
  <NumberOfPeriods>0x00</NumberOfPeriods>
  <EndTime>0x00000000</EndTime>
  <IntervalChannel>{enumeration}</IntervalChannel>
</Command>
```

Element	Range	Description
MeterMacId	0xFFFFFFFFFFFFFFFF; Optional if only one meter	Unique MAC Address of meter
NumberOfPeriods	0x0 – 0xc	Number of intervals requested; maximum is 12.
EndTime	0x0 – 0xFFFFFFFF	UTC time of the end of the most chronologically recent interval; 0x0 indicates the most recent interval block
IntervalChannel	Delivered | Received	Delivered: Interval data for commodity delivered by the utility to the user. Received: Interval data for commodity received by the utility from the user.

12. Notify: *ProfileData*

The **RAVEn™** sends the *ProfileData* notification in response to the GET_PROFILE_DATA command. It provides a series of interval data as recorded by the meter. The interval data was captured with a periodicity specified by the ProfileIntervalPeriod field. The content of the interval data depends on the type of information requested using the IntervalChannel field in the GET_PROFILE_DATA command. Data is organized in reverse chronological order: the most recent interval is transmitted first and the oldest interval is transmitted last.

```
<ProfileData>
  <DeviceMacId>0xFFFFFFFFFFFFFFFF</DeviceMacId>
  <MeterMacId>0xFFFFFFFFFFFFFFFF</MeterMacId>
  <EndTime>0x00000000</EndTime>
  <Status>0x00</Status>
  <ProfileIntervalPeriod>0</ProfileIntervalPeriod>
  <NumberOfPeriodsDelivered>0x00</NumberOfPeriodsDelivered>
  <IntervalData>0x000000</IntervalData>
  [ … ]
</ProfileData>
```

Element	Range	Description
DeviceMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of the **RAVEn™**
MeterMacId	0xFFFFFFFFFFFFFFFF	Unique MAC Address of meter
EndTime	0x0 – 0xFFFFFFFF	UTC time of the end of the most chronologically recent interval requested
Status	0x0 – 0x05	Status of returned data: 0x00 Success 0x01 Undefined Interval Channel requested 0x02 Interval Channel not supported 0x03 Invalid End Time 0x04 More periods Requested than can be returned 0x05 No intervals available for the requested time
ProfileIntervalPeriod	0 - 7	The length of each sampling interval: 0 Daily 1 60 minutes 2 30 minutes 3 15 minutes 4 10 minutes 5 7.5 minutes 6 5 minutes 7 2.5 minutes
NumberOfPeriodsDelivered	0x0 – 0xFF	The number of intervals being returned.
IntervalData	0x0 – 0xFFFFFF	Series of interval data from the meter. Most recent interval is first; oldest is last. Invalid intervals are marked as 0xFFFFFF.