

# Isadore RESTian API

Please note, I'm trying to move this documentation into RAML. The documentation in RAML may not be complete but lets get it there.

[Isadore RESTian API](#)

[Introduction](#)

[Login](#)

[Account Recovery](#)

[Contact](#)

[Accounts](#)

[Settings Management](#)

[Data Access](#)

[Reports](#)

[Generic Reports](#)

[Custom Reports](#)

[Alarms](#)

[Controls](#)

[Look Up Tables](#)

[Change Log](#)

## Introduction

This document exists as an effort to describe the RESTian API for the Isadore system. For a introduction to REST see [Building Web Services the REST Way](#) by Roger L. Costello, [Learn REST: A Tutorial](#) by Dr. M. Elkstein, [Representational State Transfer](#) Wikipedia article, and [REST Patterns](#).

Noted REST conventions.

- State Transfer - REST calls always pass full state information on each request. Use HTTP auth if authentication is needed. An authentication token will be sent through the HTTP authentication header with every request.
- Caching - Cache headers need to be correct on if a resource should be cached or when it should expire.
  - Cache-Control: private Etag: hash composed of(userid, contents?)
- Information is organized by URL `http://site.com/resource/identifier`
- Method uses:
  - GET -- Read only calls for resource
  - POST -- Create resource
  - PUT -- Replace resource
  - DELETE -- Remote resource

- Only GET Methods can be cached.
- Static content:
  - Cache-Control: public Etag: hash
- Check for authentication prior to sending any other response.
- User types
  - super-user - This is us, Dan and Russell
  - power-user - Someone that could make more users and set privilege up to power user. They can also create and modify alarms for others.
  - config-user - someone that can configure the stuff but not mess with user accounts/alarms
  - fill-user - Can update fill information
  - user - Someone with read only on everything.

## Authentication

It can be done in several ways

- HTTP Basic Auth
- Cookies get set after a successful request to /resources/login, named exemail/exrhash

Both cookies and basic auth could take email/password or email/remember\_hash.

See: [browser problems with HTTP Basic Auth and logouts](#).

## Parameter Formats

- datetime parameters - [ISO8601 format](#) with timezone or 'now' for current server time.  
Example: First of February 1945 6:33:00pm would be **1945-02-01T18:33:00-06:00** It doesn't matter which time zone offset is used as long as it represents the actual time that is wanted.  
Note: datetimes are returned in json also in ISO8601 format.
- array parameters - comma separated.

## Color legends:

- **Green** - Done and tested with unit test code.
- **Yellow** - Implemented with no unit test code.
- **Red** - Unimplemented changes.
- **Orange** - Low priority, can wait to implement.

## Login

**URL:** /resources/login

**Method:** POST

Authentication Required: User

Description: Check credentials/sets cookie, should have anti-brute force feature (delay between each try).

User Story Tie-in: U1

Parameters:

- remember - 'true' to enable the remember me functionality

Response:

- 204 - On successful login
- 401 - Unauthorized (wrong/no credentials)

**URL:** /resource/login/logout

**Method:** PUT

Authentication Required: None

Description: Logs the user out

User Story Tie-in: U1-3

Parameters: None

Response

- 204 - Successful logout

## Account Recovery

**URL:** /resources/accounts/recover

User story tie-in: U1-2

**Method:** POST

Authentication Required: None

Description: Sends recover email. Limit to around 30 tries an hour.

Parameters

- email

Response

- 204 Response successful
- 400 Email not in DB (is this abusable?)
- 400 all requests once maximum is reached.

**Method:** PUT

Authentication Required: None

Description: sets password after following recover link. (Should have brute force protections)

Parameters

- email
- recovery\_hash - The recovery hash
- new\_password

Response

- 200 Success email changed
- 400 Bad request (missing password)
- 401 Unauthorized (Wrong email/recover\_hash or too many failed attempts)

## Contact

**URL:** /resources/contact

**Method:** POST

Authentication Required: user+

Description: Send message to EA staff.

Parameters:

- email
- name
- subject
- message

Response

- 202 - Message sent
- 401 - unauthorized
- 400 bad arguments (missing message)

## Accounts

**URL:** /resources/privileges

**Method:** GET

Authentication Required: user+

Description: Get index of all privilege types

Parameters

- format - xml|json

Response

- 200 - xml|json of urls to all privileges
- 401 Unauthorized

**URL:** /resources/privileges/[id]

**Method:** GET

Authentication Required: user+

Description: Get privilege information.

Parameters

- format - xml|json

Response

- 200 - xml|json of content
- 401 Unauthorized
- 404 privilege id does not exist

**URL:** /resources/accounts

User story tie-in: U5

Authentication Required: power-user+

**Method:** GET

Authentication Required: power-user+ or own id

Description: Gets a index of all accounts, is xlinks (for XML type) or list of resources pointing to all user accounts at current auth privilege or lower.

Parameters

- format - xml|json

Response

- 200 - xml|json of urls to all accounts
- 401 Unauthorized

Method: POST

Authentication Required: power-user+

Description: Create an account and send email to new account email.

Parameters

- format - xml|json
- name
- email
- phone
- privilege\_id

Response

- 200 - XML|JSON of resource url corresponding to created user.
- 400 - Missing/bad arguments (account not created).
- 401 - Unauthorized

**URL:** /resources/accounts/[id]

User story tie-in: U5, U6

Method: GET

Authentication Required: user+

Description: Gets viewable user information.

Parameters

- format - xml|json

Response

- 200 - XML|JSON of user details
- 401 - Unauthorized
- 404 - User ID doesn't exist (check authorized first).

Method: DELETE

Authentication Required: power-user+

Description: Sets enabled\_p flag to FALSE.

Response

- 204 - Success in deletion
- 401 - Unauthorized
- 404 - user doesn't exist

Method: PUT

Authentication Required: power-user+ or self

Description: Update account info

#### Parameters

- format - xml|json
- name
- email
- phone
- password (hashed)
- privilege - power-user+ only

#### Response

- 204 - Account updated
- 400 - Missing/bad arguments
- 401 - Unauthorized

**URL:** /resources/accounts/self

**Method:** GET

Description: Gives your url of the account you are authenticating as.

#### Parameters

#### Response

- 200 - xlink to account url
- 401 - unauthorized

**URL:** /resources/accounts/self-fast

**Method:** GET

Description: Gives your the account you are authenticating as.

#### Parameters

#### Response

- 200 - the account data
- 401 - unauthorized

## Settings Management

**URL:** /resources/conf/version

**Method:** GET

Authentication Required: user+/MID

Description: Gets current conf version number, this number is incremented everytime there is a modification involving anything in conf (Sensors, devices, bins, types, etc).

#### Parameters

#### Response

- 200 - json - 'conf\_version' a integer
- 401 unauthorized

**URL:** /resources/conf/general

User story tie-in: U9-1

**Method:** GET

Authentication Required: user+/MID

Description: Gets current general settings.

Parameters

- format - xml|json

Response

- 200 - xml|json containing data
- 401 Unauthorized

**Method:** PUT

Authentication Required: config-user+

Description: Sets general configuration

Parameters

- format - xml|json
- frequency - value in seconds
- enabled\_p
- mid\_pass (requires super-user)
- alarm\_clear\_interval (in minutes)

Response

- 204 - success
- 401 - unauthorized
- 400 - bad request

**URL:** /resources/conf/bins-fast

**Method:** GET

Authentication Required: user+

Description: Gets all bins.

Parameters:

Response:

- 200 - bins data
- 401 - Unauthorized.

**URL:** /resources/conf/bins

**Method:** GET

Authentication Required: user+

Description: Get url list of all bins.

Parameters:

- format: xml|json

Response:

- 200 - xml|json containing data
- 401 - Unauthorized.

**Method:** POST

Authentication Required: config-user+

Description: Create new bin

Parameters:

- format - xml|json
- name
- x
- y

Response

- 200 - success with url to new bin
- 401 - unauthorized
- 400 - bad request

**URL:** /resources/conf/bins/[id]

**Method:** GET

Authentication Required: user+

Description: Get bin info.

Parameters:

- format - xml|json

Response:

- 200 - xml|json containing data
- 401 - Unauthorized.
- 404 - Bin does not exist.

**Method:** PUT

Authentication Required: config-user+

Description: Update a bin

Parameters:

- format - xml|json
- name
- x
- y

Response

- 204 - success
- 400 - bad request
- 401 - unauthorized
- 404 - bin does not exist

**Method:** DELETE

Authentication Required: config-user+

Description: Remove bin

Parameters:

Response:

- 204 - Success
- 401 - Unauthorized



- 404 - Bin does not exist

**URL:** /resources/conf/bin\_sections-fast

**Method:** GET

Authentication Required: user+

Description: Gets all bin sections.

Parameters:

Response:

- 200 - bin sections data
- 401 - Unauthorized

**URL:** /resources/conf/bin\_sections

**Method:** GET

Authentication Required: user+

Description: Get url list of all bin sections.

Parameters:

- format: xml|json

Response:

- 200 - xml|json containing data
- 400 - bad arguments
- 401 - Unauthorized

**Method:** POST

Authentication Required: config-user+

Description: Create a new bin section.

Parameters:

- format - xml|json
- name

Response

- 200 - success with url to new bin section
- 400 - bad arguments
- 401 - unauthorized

**URL:** /resources/conf/bin\_section/[id]

**Method:** GET

Authentication Required: user+

Description: Get bin section info.

Parameters:

- format - xml|json
- bin\_id - need new table if we are doing this bin\_to\_bin\_section

Response:

- 200 - xml|json containing data
- 401 - Unauthorized.
- 404 - Bin section does not exist.

**Method:** PUT

Authentication Required: config-user+

Description: Update a bin section

Parameters:

- name

Response

- 204 - success
- 400 - bad request
- 401 - unauthorized
- 404 - bin section does not exist

**Method:** DELETE

Authentication Required: config-user+

Description: Remove bin section

Parameters:

Response:

- 204 - Success
- 401 - Unauthorized
- 404 - Bin section does not exist

**URL:** /resources/conf/sensor\_types-fast

**Method:** GET

Authentication Required: user+

Description: Gets all sensor types

Response:

- 200 - sensor types data
- 401 - Unauthorized

**URL:** /resources/conf/sensor\_types

**Method:** GET

Authentication Required: user+

Description: Get url list of all sensor types

Parameters:

- format: xml|json
- device\_type\_id

Response:

- 200 - xml|json containing data
- 400 - bad arguments
- 401 - Unauthorized

**URL:** /resources/conf/sensor\_types/[id]

**Method:** GET

Authentication Required: user+

Description: Get sensor types info

#### Parameters

- format - xml|json

#### Response

- 200 - xml|json of data
- 401 - unauthorized
- 404 - sensor\_types with id not found.

**URL:** /resources/conf/device\_types-fast

**Method:** GET

Authentication Required: user+

Description: Gets all of device types

#### Parameters

#### Response

- 200 - device types data
- 401 - unauthorized

**URL:** /resources/conf/device\_types

**Method:** GET

Authentication Required: user+

Description: List of device types

#### Parameters

- format - xml|json

#### Response

- 200 - xml|json list of urls
- 401 - unauthorized

**URL:** /resources/conf/device\_types/[id]

**Method:** GET

Authentication Required: user+

Description: Get info about the device type

#### Parameters

- format - xml|json

#### Response:

- 200 - xml|json of device\_type info
- 401 - unauthorized
- 404 - device\_type not found

**URL:** /resources/conf/devices

User story tie-in: U9-2, U9-4, U9-5

**Method:** GET

TODO: We have no way to get a list of valid years, year should be optional?

Authentication Required: user+/MID

Description: Lists existing devices

Parameters

- format - xml|json
- bin\_id - optional, only list devices in certain bin
- bin\_section\_id - optional, only list devices in certain bin
- year - required

Response

- 200 - xml|json list of urls
- 400 - bad arguments
- 401 - unauthorized

**Method:** POST

Authentication Required: config-user+

Description: Add a device

Parameters

- device\_type\_id
- port
- address
- enabled\_p
- bin\_id
- bin\_section\_id
- name
- year

Response

- 200 - success with id of new item
- 401 - unauthorized
- 400 - bad request

**URL:** /resources/conf/devices-fast

**Method:** GET

Authentication Required: user+/MID

Description: gets data of devices (compared to xlink version of nonfast) also includes sensors and sensor\_mirror in the data.

Parameters

- format - xml|json
- bin\_id - optional, only list devices in certain bin
- bin\_section\_id - optional, only list devices in certain bin
- year - optional, will use current server year if missing.

Response

- 200 - xml|json list of urls
- 400 - bad arguments
- 401 - unauthorized

**URL:** /resources/conf/devices/[device\_id]

User story tie-in: U9-2, U9-4, U9-5

**Method:** GET

Authentication Required: user+/MID

Description: Returns device info

Parameters

- format - xml|json

Response

- 200 - xml|json of device info.
- 401 - Unauthorized
- 404 - Device ID not found

**Method:** PUT

Authentication Required: config-user+

Description: Set values for pre-existing device

Parameters

- device\_type\_id
- port
- address
- enabled\_p
- bin\_id
- bin\_section\_id
- name
- year

Response

- 204 - success
- 400 - bad request
- 401 - unauthorized
- 404 - device doesn't exist

**Method:** DELETE

Authentication Required: config-user+

Description: Remove device

Response:

- 204 - Success
- 401 - unauthorized
- 404 - device doesn't exist

**URL:** /resources/conf/devices\_clone

**Method:** POST

Authentication Required: Config+

Description: Clones device configuration from one year into another

Parameters

- from\_year

- to\_year

#### Response

- 204 - clone successful
- 401 - unauthorized
- 400 - bad parameters

**URL:** /resources/conf/sensors

**Method:** GET

Authentication Required: user+/MID

Description: Get a list of all sensors or sensors in a device.

#### Parameters

- format - xml|json
- device\_id - optional otherwise gets list of all sensors.

#### Response

- 200 - url of sensors
- 400 - bad request
- 401 - unauthorized

**Method:** POST

Authentication Required: config-user+

Description: Create new sensor

#### Parameters

- format - xml|json
- device\_id - device this sensor is on
- sensor\_type\_id - sensor type
- convert\_py - conversion math function
- bias - bias factor
- enabled\_p

#### Response

- 200 - url of new sensor
- 400 - bad request
- 401 - unauthorized

**URL:** /resources/conf/sensors/[id]

**Method:** GET

Authentication Required: user+

Description: Get sensor info

#### Parameters

- format - xml|json

#### Response

- 200 - xml|json of info
- 401 - unauthorized
- 404 - sensor does not exist.

**Method:** PUT

Authentication Required: config-user+

Description: Update sensor

Parameters

- format - xml|json
- device\_id - device this sensor is on
- sensor\_type\_id - sensor type
- convert\_py - conversion math function
- bias - bias factor
- enabled\_p

Response

- 204 - Success
- 400 - bad arguments
- 401 - unauthorized
- 404 - sensor not found

**Method:** DELETE

Authentication Required: config-user+

Description: Remove sensor (This probably will be never used, but here since we can add one.

Sensors could probably get removed when device is.)

Response:

- 204 - Success
- 401 - unauthorized
- 404 - device doesn't exist

**URL:** /resources/conf/sensors-fast

**Method:** GET

Authentication Required: user+

Description: Get all sensors belonging to a device

Parameters

- device\_id

Response

- 200 - json data of sensors
- 401 - unauthorized
- 400 - bad arguments

**Method:** DELETE

Authentication Required: config-user+

Description: Delete all sensors belonging to device

Parameters

- device\_id

Response

- 204 - sensors deleted
- 401 - unauthorized
- 400 - bad arguments

**URL:** /resources/conf/sensor\_mirrors

**Method:** POST

Authentication Required: config-user+

Description: New sensor mirror

Parameters

- device\_id - The device it belongs to
- sensor\_type\_id - The sensor type to mirror
- bin\_id - The bin to mirror
- bin\_section\_id - The bin section to mirror

Response

- 200 - url of new sensor mirror
- 401 - unauthorized
- 400 - bad arguments

**URL:** /resources/conf/sensor\_mirrors/[id]

**Method:** GET

Authentication Required: user+

Description: Get sensor mirror data

Response

- 200 - json of sensor mirror
- 401 - unauthorized

**Method:** DELETE

Authentication Required: config-user+

Description: Delete sensor mirror

Response

- 204 - sensor mirror deleted
- 401 - unauthorized
- 404 - sensor mirror doesn't exist

**URL:** /resources/conf/sensor\_mirrors-fast

**Method:** GET

Authentication Request: user+

Description: Get all sensor mirrors for a device

Parameters

- device\_id

Response

- 200 - json for all sensor\_mirrors for a device
- 401 - unauthorized
- 400 - bad argument

**Method:** DELETE

Authentication Request: config-user+

Description: Delete all sensor\_mirrors belonging to a device

Parameters



- device\_id - The device to delete all sensor mirrors from.

Response

- 204 - sensor mirrors for device deleted
- 401 - unauthorized

**URL:** /resources/conf/subsamples

**Method:** GET

Authentication Required: user+

Description: Get list of precalculated subsamples.

Response

- 200 - json data
- 401 Unauthorized

## Data Access

**URL:** /resources/data/readings/last

User story tie-in: U7

**Method:** GET

Authentication Required: user+

Description: URL of the latest reading from the most recent /resources/data/readings POST

Parameters

- format - xml|json

Response

- 200 - xml|json - 'xlink' , 'datetime', 'countdown'
- 401 - unauthorized
- 404 - last reading not found
  - Could only happen when readings table is empty?

**URL:** /resources/data/readings

User story tie-in: U7, U8

**Method:** POST

Authentication: MID

Description: Save new data from sensors (Used by MID)

Parameters

- time
- data\_count - number of values being reported
- sensor\_id\_x - sensor id this data is for
- sensor\_raw\_x - sensor raw value

Response

- 204 - successfully saved

- 400 - bad request/arguments
- 401 - unauthorized

**URL:** /resources/data/readings/[reading\_id]

User story tie-in: U7, U8

**Method:** GET

Authentication Required: user+

Description: Get data for a reading.

Parameters

- format - xml|json
- bin\_id - id
- bin\_section\_id
- sensor\_type\_id

Response

- 200 - xml|json data in 'results' even if one row.
- 401 - unauthorized
- 404 - data for date does not exist

**URL:** /resources/data/sensor\_data\_latest

**Method:** GET

Authentication Required: user+

Description: Get latest sensor data for specific sensors

Parameters:

- sensor\_ids - ARRAY

Response

- 200 - json data in 'results'
- 401 - unauthorized
- 400 - bad request/arguments

**URL:** /resources/data/fill\_number

User story tie-in: U10

**Method:** GET

Authentication Required: user+

Description: Map from fill numbers to fill ids.

Parameters

- format - xml|json
- fill\_number
- begin\_datetime
- end\_datetime

Response

- 200 - xml|json list of fill urls
- 400 - bad arguments
- 401 - unauthorized

- 404 - nothing found

**URL:** /resources/data/fill\_types

Method: GET

Authentication Required: user+

Description: Get fille type data

Parameters: None

Response

- 200 - JSON of fill\_types
- 401 - unauthorized

**URL:** /resources/data/fills-fast

User story tie-in: U10

**Method: GET**

Authentication Required: user+

Description: Get fills data

Parameters

- bin\_id - optional
- begin\_span1 - Find fills starts between two dates. [span1, span2]
- begin\_span2

Response

- 200 - fills data
- 400 - bad arguments
- 401 - unauthorized

**URL:** /resources/data/fills

User story tie-in: U10

**Method: GET**

Authentication Required: user+

Description: Get fill urls matching criteria in parameters

Parameters (If search a column, columns that are null will always be false, specifying a end\_datetime when one hasn't been set will result in a false.)

- format - xml|json
- bin\_id
- field\_code
- hybrid\_code
- storage\_bin\_number
- storage\_bin\_code
- begin\_span1 - Find fills starts between two dates. [span1, span2]
- begin\_span2
- end\_span1 - Find fills that end between two dates [span2, span2]
- end\_span2

- begin\_datetime - Find where begin\_datetime is >= given date
- end\_datetime - Find where end\_datetime is <= given date
- during\_datetime - Find fills that are happening at a given time period, takes into account null end\_datetime as ongoing.

Response

- 200 - xml|json list of fill urls, if nothing found, empty xlink
- 400 - bad arguments
- 401 - unauthorized

#### Method: POST

Authentication: fill-user+ (Should this be user+, or new class like meta-user?)

Description: New metadata set.

Parameters

- fill\_number
- filled\_datetime
- bin\_id

Response

- 200 - xml|json id created
- 400 - bad arguments
- 401 - unauthorized

**URL:** /resources/data/fills/[fill\_id]

User story tie-in: U10

#### Method: GET

Authentication Required: user+

Description: Get some fill data

Parameters:

- format - json|xml

Response:

- 200 - fill data
- 401 - unauthorized
- 404 - fill not found

#### Method: PUT

Authentication Required: fill-user+ (Should this be user+, or new class like meta-user?)

Description: Update fill.

Parameters

- fill\_number - \*Required
- filled\_datetime - datetime \*Required
- emptied\_datetime - datetime
- air\_begin\_datetime - datetime
- air\_end\_datetime - datetime
- rotation\_number
- bin\_id
- hybrid\_code

- field\_code
- storage\_bin\_number
- storage\_bin\_code
- pre\_mc - array
- post\_mc - array
- bushels
- roll\_datetime - array of datetimes
- mc\_maxtemp\_lut\_id - empty for system default

Response

- 204 - success
- 400 - bad arguments
- 401 - unauthorized
- 404 - fill not found

#### Method: DELETE

Authentication Required: fill-user+ (Should this be user+, or new class like meta-user?)

Description: Remove the fill

Response:

- 204 - success
- 401 - unauthorized
- 404 - fill not found

**URL:** /resource/data/air\_deductions-fast

#### Method: GET

Authentication Required: user+

Description: Get all air deduction data

Parameters:

- begin\_span1 - optional
- end\_span2 - optional

Response:

- 200 - air deductions
- 401 - unauthorized

**URL:** /resources/data/air\_deductions

#### Method: GET

Authentication Required: user+

Description: Gets xlink data to deductions

Parameters:

- begin\_span1 - optional
- end\_span2 - optional

Response:

- 200 - xlink data
- 401 - unauthorized

#### Method: POST

Authentication Required: fill user+  
Description: Add new air deduction  
Parameters:

- begin\_datetime
- end\_datetime

Response

- 200 - xlink of new air duction
- 401 - unauthorized
- 400 - bad arguments

**URL:** /resources/data/air\_deductions/[id]

**Method: GET**

Authentication Required: user+  
Description: get a single air deduction record  
Response

- 200 - air deduction data
- 401 - unauthorized
- 404 - air deduction not found

**Method: PUT**

Authentication Required: fill user+  
Description: Update air\_deduction  
Parameters

- begin\_datetime
- end\_datetime

Response

- 202 - air deduction updated successfully
- 400 - bad arguments
- 404 - air deduction not found.

**Method: DELETE**

Authentication Required: fill user+  
Description: delete an air deduction  
Response

- 202 - air deduction deleted
- 404 - air deduction not found

## Reports

### Generic Reports

**URL:** /resources/data/report/datetime\_csv

User story tie-in: U8-1,

Method: GET

Authentication Required: user+

Description: Create and save downloadable CSV of data from specified time period.

Parameters:

- start\_time
- end\_time

Response:

- 200 - URL of CSV location
- 400 - bad arguments
- 401 - unauthorized

**URL:** /resources/data/graph

User story tie-in: U8-3, U8-4

Method: GET

Description: Get image of data

Parameters:

- format - png|pdf|eps|???
- begin\_datetime
- end\_datetime
- bin\_ids - Array - Or should it be limited to just one bin?
- fill\_number (replaces start and end times and bin\_id)
- bin\_section\_ids - Array (each bin section uses a different line style)
- sensor\_type\_id\_a (each sensor type uses a different color)
- sensor\_type\_id\_b (optional)

Response:

- 200 - URL to graph (or graph data, whatever is easier)
- 400 - bad arguments
- 401 - unauthorized

**URL:** /resources/data/graph/selection/[bin\_id]

Method: GET

Authentication Required: user+

Description: Get graph selection possibilities for a bin of form:

{options: ['section1', [ ['datalabel', 'request\_keyword', [request\_data]], ...], ...]}

Response:

- 200 - JSON of categorized graph selection possibilities.
- 404 - non-existent bin\_id.
- 401 - Unauthorized

**URL:** /resources/data/graph/data

Method: GET

Authentication Required: user+

Description: Get data needed for client side graphing.

Parameters:

- bin\_id
- query - type1, id1-1, id1-2; type2, id2-1, id2-2; ...  
type: sensor - id1 = bin\_section\_id, id2 = sensor\_type\_id  
type: inlet - id1 = sensor\_type\_id  
type: outlet - id1 = sensor\_type\_id  
type: maxtemp
- sample\_period - should be a value from conf/subsamples
- begin\_datetime
- end\_datetime

Response:

- 200 - CSV of data
- 400 - bad arguments
- 401 - Unauthorized

## Custom Reports

**URL:** /resources/data/report/fill\_report

User story tie-in: ???

Method: GET

Authentication Required: user+

Description:

Parameters:

- year
- begin\_fill\_number
- end\_fill\_number
- type - fill, hybrid
- form - full, tab, csv
- display\_tz - timezone name (America/Chicago)

Response:

- 200 - URL
- 400 - bad arguments
- 401 - unauthorized

## Alarms

**URL:** /resources/alarms

User story tie-in: U9

Method: GET

Authentication Required: user+



Description: Get a list of links to alarms that exist for the user.

Parameter:

- format - json|xml
- account\_id

Response:

- 200 - xml|json of links to alarms
- 401 - unauthorized

#### Method: POST

Authentication Required: user+

Description: Create new alarm

Parameters:

- alarm\_type\_id
- account\_id - must be own account\_id or are power-user+
- greater\_than\_p
- alarm\_contact\_type\_ids - ARRAY of contact types
- value

Response:

- 200 - URL for new alarm
- 400 - bad arguments
- 401 - unauthorized

**URL:** /resources/alarms/[alarm\_id]

User story tie-in: U-9

#### Method: GET

Authentication Required: user+

Description: Gets alarm details

Parameters

- format - json|xml

Response

- 200 - json|xml of data
- 401 - unauthorized
- 404 - no alarm of that id

#### Method: PUT

Authentication Required: power-user+ or own alarm

Description: Update alarm details. Non-power-users cannot change alarm details for alarms that other users are receiving notifications for.

Parameters:

- (See POST: /resource/alarms)

Response:

- 204 - success
- 400 - bad arguments
- 401 - unauthorized

- 404 - alarm not found

**Method: DELETE**

Authentication Required: power-user+ or own alarm

Description: Delete alarm. Non-power-users cannot delete alarms for alarms that other users are receiving notifications for.

Parameters:

Response:

- 204 - success
- 401 - unauthorized
- 404 - alarm not found

**URL:** /resources/alarms/alarm\_types-fast

**Method: GET**

Authentication Required: user+

Description: Gets available alarm types (e.g. bin temp, fan temp).

Parameters:

- format - xml|json

Response:

- 200 - alarm\_types data
- 401 - unauthorized

**URL:** /resources/alarms/alarm\_types

**Method: GET**

Authentication Required: user+

Description: Get a list of available alarm types (e.g. bin temp, fan temp).

Parameters:

- format - xml|json

Response:

- 200 - xml|json list of alarm\_types url
- 401 - unauthorized

**URL:** /resources/alarms/alarm\_types/[id]

**Method: GET**

Authentication Required: user+

Description: Get alarm\_type info

Parameters:

- format - xml|json

Response:

- 200 - the info
- 401 - unauthorized
- 404 - not found

**URL:** /resources/alarms/alarm\_contact\_types-fast

**Method: GET**

Authentication Required: user+

Description: Gets available alarm contact types (SMS, email, etc).

Parameters:

Response:

- 200 - all alarm contact types
- 401 - unauthorized

**URL:** /resources/alarms/alarm\_contact\_types

**Method: GET**

Authentication Required: user+

Description: Get a list of available alarm contact types (SMS, email, etc).

Parameters:

- format - xml|json

Response:

- 200 - xml|json list of contact\_types url
- 401 - unauthorized

**URL:** /resources/alarms/alarm\_contact\_types/[id]

**Method: GET**

Authentication Required: user+

Description: Get contact\_type info

Parameters:

- format - xml|json

Response:

- 200 - the info
- 401 - unauthorized
- 404 - not found

**URL:** /resources/alarms/events

User story tie-in: U8-3

**Method: GET**

Authentication Required: user+

Description: Get list of links to when alarm events took place

Parameters

- format - json|xml
- begin - MM/DD/YYYY HH:MM:SS
- end - MM/DD/YYYY HH:MM:SS

Response:

- 200 - json|xml of data of links to alarm events
- 401 - unauthorized
- 404 - bad arguments

**URL:** /resources/alarms/events/[event\_id]

User story tie-in: U8-3

**Method:** GET

Authentication Required: user+

Description: Get details about an alarming turn of events that took place.

Parameters

- format - json|xml

Response:

- 200 - json|xml of data
- 401 - unauthorized
- 404 - alarm event not found.

**URL:** /resources/alarms/globalEvents

User story tie-in: U8-3

**Method:** GET

Authentication Required: user+

Description: Get list of links to when alarm events took place

Parameters

- begin - MM/DD/YYYY HH:MM:SS
- end - MM/DD/YYYY HH:MM:SS

Response:

- 200 - json|xml of data of links to alarm events
- 401 - unauthorized
- 404 - bad arguments

**URL:** /resources/alarms/globalEvents/[event\_id]

User story tie-in: U8-3

**Method:** GET

Authentication Required: user+

Description: Get details about an alarming turn of events that took place.

Parameters

- format - json|xml

Response:

- 200 - json|xml of data
- 401 - unauthorized
- 404 - alarm event not found.

**URL:** /resources/alarms/currentEvents

Method: GET

Authentication Required: user+

Description: Gets xlinks to all ongoing alarm events available to the user (global or user

alarms).

Response:

- 200 - json xlink urls (could be links to global events or events)
- 401 - unauthorized

URL: /resources/alarms/pastEvents

Method: GET

Authentication Required: user+

Description: Gets xlink to all alarm events that are not currently ongoing (global or user alarms).

Parameters

- after\_datetime -

Response:

- 200 - json xlink urls
- 401 - unauthorized
- 400 - bad argument

## Controls

URL: /resources/controls

Method: GET

Authentication Required: User+

Description: Gets lists of controls based on search parameters given.

Method: POST

Authentication Required: Config User+

Description: Set a control.

Parameters

- sensor\_id
- value

Response

- 200 - xlink to newly created control set.
- 401 - unauthorized
- 400 - bad sensor, sensor not controllable, or bad value given.

URL: /resources/controls/last

Method: GET

Authentication Required: User+/MID

Description: Get xlink for current set control for sensor

Parameters

- sensor\_id

Response

- 200 - xlink of last control for sensor

- 401 unauthorized
- 400 - sensor doesn't exist, or not controllable.

**URL:** /resources/controls/id

**Method:** GET

Authentication Required: User+/MID

Description: Get details about last set control

Response

- 200 - The control set value info
- 401 - unauthorized
- 404 - control set not found.

**URL:** /resources/controls/id

**Method:** PUT

Authentication Required: User+/MID

Description: Get details about last set control

Response

- 200 - The control set value info
- 401 - unauthorized
- 404 - control set not found.

## Look Up Tables

**URL:** /resources/luts/mc\_maxtemp\_default

**Method:** PUT

Authentication Required: config-user+

Description: Sets the default mc\_maxtemp lut

Parameters

- mc\_maxtemp\_lut\_id

Response

- 203 - default set
- 401 - unauthorized
- 400 - bad arguments

**URL:** /resources/luts/mc\_maxtemp-fast

**Method:** GET

Authentication Required: User+

Description: Get all mc\_maxtemp luts

Response

- 200 - The mc\_maxtemp luts
- 401 - unauthorized

**URL:** /resources/luts/mc\_maxtemp

**Method:** POST

Authentication Required: config-user+

Description: Make new mc maxtemp lut

Parameters

- name
- mc\_per\_hour
- mcs - array comma seperated, should match up with maxtemps
- maxtemps - array comma seperated, should match up with mcs

Response

- 200 - xlink to new lut
- 401 - unauthorized

**URL:** /resources/luts/mc\_maxtemp/[mc\_maxtemp\_id]

**Method:** PUT

Authentication Required: config-user+

Description: Update lut

Parameters

- name
- mc\_per\_hour
- mcs - array comma seperated, should match up with maxtemps
- maxtemps - array comma seperated, should match up with mcs

Response

- 401 - Unauthorized
- 404 - no lut with id found

**URL:** /resources/luts/mc\_maxtemp/[mc\_maxtemp\_id]

**Method:** DELETE

Authentication Required: config-user+

Description: Remove a mc maxtemp lut

Response

- 203 -deleted
- 401 - Unauthorized
- 404 - no lut with that id found

## Change Log

- 6/18/2014 - change datetime format.