

SmartConnector RESTful EWS Gateway

Buildings Labs

- 1 Support..... 3
- 2 Revision History 3
- 3 Overview 4
 - 3.1 Scope..... 4
 - 3.2 What is SmartConnector RESTful EWS Gateway? 4
- 4 REST Extension Framework..... 6
 - 4.1 Authentication 6
 - 4.2 RESTful URLs..... 6
 - 4.3 Self- Documenting..... 6
 - 4.4 Filtering, Sorting, and Paging 6
 - 4.5 HTTP Status Codes 7
 - 4.6 Response Headers..... 7
 - 4.7 Layered System 7
 - 4.8 Rate Limiting 8
- 5 RESTful EWS Gateway 9
 - 5.1 Data Providers..... 9
 - 5.1.1 MongooseEwsRestProvider 9
 - 5.1.2 SoapEwsRestProvider 9
 - 5.1.3 SboEwsRestProvider 9
 - 5.2 Provisioning and Configuring Endpoints 9
 - 5.2.1 Adding REST Endpoint..... 9
 - 5.2.2 Configuring REST Endpoint 10
 - 5.2.3 Complete Configuration..... 14
 - 5.3 Resources and Actions 14
 - 5.3.1 AlarmEvents 14
 - 5.3.2 AlarmEventTypes 18
 - 5.3.3 Alarms 21
 - 5.3.4 Containers 29
 - 5.3.5 Notifications..... 35
 - 5.3.6 Root..... 38

- 5.3.7 Subscriptions..... 39
- 5.3.8 SystemEvents 42
- 5.3.9 SystemEventTypes 45
- 5.3.10 Trends 47
- 5.3.11 TrendSamples..... 52
- 5.3.12 Values..... 55
- 6 Swagger..... 65
 - 6.1 Using Swagger 66
 - 6.1.1 Authentication 66
 - 6.1.2 Issuing Requests..... 67
 - 6.1.3 Parameters..... 69
 - 6.2 Metadata and Client Code Generators 76
- 7 Security Considerations 76

1 Support

Schneider Electric provides branch and channel partners with planning and implementation assistance for Buildings Labs. To request help, send an email to Product Support specifying the solution name and the type of assistance you require. Product Support will relay your request to the appropriate support team.

North America (NAM) Product Support

Building Management Systems (BMS): productsupport.NAM-BMS@schneider-electric.com

Global Product Support

Building Management Systems (BMS): productsupport.BMS@schneider-electric.com

2 Revision History

Date	Author	Revision	Changes Made
2/21/2017	MRS	1	Initial release.
12/15/2017	MRS	2	Updated for v2.3
08/16/2018	JMB	3	Updated for v2.4

© 2017 Schneider Electric. All Rights Reserved. Schneider Electric, StruxureWare, SmartStruxure solution, and EcoStruxure are trademarks owned by Schneider Electric Industries SAS or its affiliated companies. All other trademarks are the property of their respective owners.

3 Overview

3.1 Scope

This document is intended as a guide for developers wishing to exchange data via a SmartConnector enabled REST endpoint using an API and data model based on the Schneider Electric EcoStruxure Web Services (EWS) data model. This document assumes the reader is familiar with the [EcoStruxure Web Service Specification](#), has the requisite knowledge in consuming a RESTful HTTP endpoint, and is sufficiently proficient in the programming language they are developing in.

This document will not cover the details pertaining to installation, monitoring, and control of SmartConnector in a runtime environment. That information can be found in *SmartConnector Installation and Configuration Guide*. Configuration instructions, as they pertain to REST endpoints will be covered.

Furthermore, this document will not cover the details in consuming SmartConnector's libraries for the purpose of authoring customized middleware applications (SmartConnector Extensions) to run in SmartConnector. That information can be found in *SmartConnector Developers Guide*.

3.2 What is SmartConnector RESTful EWS Gateway?

What exactly is "RESTful EWS Gateway" and how can it be used with EWS which is based on SOAP? Let's review some concepts.

Simple Object Access Protocol (SOAP) and Representational State Transfer (REST) should never be compared directly (yet they often are); SOAP is a protocol while REST is an architecture.

REST was first described by Roy Fielding in his 2000 PhD dissertation on [network based software architectures](#). From Fielding's own dissertation, the major characteristics of REST systems are: client-server based, stateless, cacheable for performance optimization, with layered abstractions.

SOAP was designed by Winer, Box, Atkinson, and Al-Ghosein for Microsoft in 1998 as a messaging protocol. It was submitted to IETF in 1999 and in 2003 (as v1.2) became a [W3C recommendation](#). SOAP has three major characteristics: extensibility, transport layer neutrality, and programming model independence. Lacking from the specification is security and routing definitions.

EWS is Schneider Electric's SOAP interface for system to system data exchange. From the EWS specification itself:

"EWS defines a common interface for the exchange of data between systems in EcoStruxure-targeted solutions. Products and systems adhering to the specification can quickly and easily exchange information such as state, historical and alarm data. EWS also supports the exchange of system information, for example, hierarchical structure and metadata. It does not, however, support the ability for one system to create new objects and data structures in another system."

The next logical question is, “If a SOAP API already exists, why do we need a RESTful one?” The honest answer is you don’t. If you are comfortable developing applications using a SOAP client against a EWS server then you can continue to do so. However, if you prefer to move away from SOAP to a more portable lightweight abstraction, which can provide full Create, Retrieve, Update, and Delete capabilities (CRUD), then this gateway is for you.

A frequently requested feature for SmartConnector was to serve EWS data in a RESTful manner. This request was followed closely by requests for SmartConnector to provide an infrastructure which allowed SmartConnector Developers to author and subsequently host custom RESTful APIs for others to consume.

SmartConnector now provides a REST Extension framework along with a specific implementation of it: the SmartConnector RESTful EWS Gateway.

4 REST Extension Framework

While Fielding does an excellent job describing what REST architecture is, there is no single standard for designing one. Rather, the internet is full of opinions on what makes the best RESTful architecture. The design approach implemented in SmartConnector was a pragmatic one based on many REST best practices. They are:

- [Authentication](#)
- [RESTful URLs](#)
- [Self-Documenting](#)
- [Filtering, Paging, and Sorting](#)
- [HTTP status codes](#)
- [Response headers](#)
- [Layered System](#)
- [Rate Limiting](#)

4.1 Authentication

RESTful APIs are stateless. This means that requests should never rely on sessions or cookies. Rather, each request should include credentials in the form of access tokens. SmartConnector REST endpoints use [OAuth2](#) and [Bearer Tokens](#) to accomplish this. Once a valid token is received from the server, it can be used in subsequent requests inside an authentication header.

4.2 RESTful URLs

REST stipulates separating your information into resources. Resources can be scalar values, logical groups of values in the form of classes, files; almost anything. These resources are then manipulated with actions using standard HTTP verbs GET, POST, PUT, PATCH, and DELETE. In this implementation the following conventions are used:

- **GET** – Retrieval of a resource.
- **POST** – Creation of a new resource.
- **PUT** – Update of a resource
- **PATCH** – Update multiple aspects of a resource at one time.
- **DELETE** – Remove a resource.

4.3 Self- Documenting

While this publication hopefully provides some useful information, it will never be as up to date as the API itself. To that end, this implementation has opted to make the API itself self-documenting using [a third party framework](#). This documentation is served directly from the configured endpoint itself.

4.4 Filtering, Sorting, and Paging

REST resource URLs should be as terse as possible while still being clear as to the resource served. Inevitably, users will desire to control what, how much, and in what order resources are returned. This

is done by means of optional query parameters. While the query parameters supported will be controller action dependent some conventions have been established.

- Filter parameters perform in logical AND fashion.
- If no overriding OrderBy parameter is provided, a default is always assumed so that resources are always returned in a deterministic order.
- Paging is supported for those actions where an unknown number of resources could be served. This is accomplished by means of Take and Skip query parameters. Take, instructs the server to return that many instances of the requested resource. Skip instructs the server to skip over that number of resources (in the implicit or explicit OrderBy) before creating the result set.

4.5 HTTP Status Codes

HTTP defines a plethora of [status codes](#). The more common ones used in this implementation are as follows:

- **200 OK** – Response to a successful request GET, PUT, or DELETE. May be used for a POST which doesn't create a resource.
- **201 Created** – Response to a successful POST which creates a resource.
- **204 No Content** – Response to a successful DELETE which removes a resource.
- **400 Bad Request** – Response to any request which is malformed.
- **401 Unauthorized** – Response to any request which does not include a valid Bearer Token.
- **404 Not Found** – Response to any request to an undefined resource.
- **429 Too Many Requests** – Response to any request which exceeds configured throttling limits.
- **500 Internal Server Error** – Response to a request which results in an error generated on the Server.

4.6 Response Headers

Response headers provide useful information which goes beyond what is included in the body of the response. The following are the response headers used in this implementation.

- **X-Total-Count** – Includes the actual number of filtered resources available for a given request when the entire set of resources is not returned.
- **Link Headers** – Link headers provide fully formed URLs to other pages of data. Link headers for the "First", "Next", and "Previous" page of resources are added when appropriate.

4.7 Layered System

Well implemented REST architectures provide layered solutions. The challenge for SmartConnector was to allow customization of these layers while still providing a usable framework. The design approach incorporated is that of a Data Provider. By abstracting the route and actions from the data, any number of "Data Providers" can authenticate and serve data from disparate sources via the same API.

4.8 Rate Limiting

Limiting the rate at which requests are processed by an API is a REST best practice. SmartConnector strongly encourages this practice by making a minimum level of “throttling” mandatory and automatic for every REST endpoint configured.

White Lists of IP address can be added to exempt throttling and overrides in the form of key based and/or IP based rates can also be added.

5 RESTful EWS Gateway

The name “RESTful EWS Gateway” was chosen to convey how this implementation relates to EWS and to show that it could be used as an alternative to SOAP based EWS communications. To be clear, this is not a port of EWS’ SOAP methods to pure HTTP. Rather, the implementation leveraged most of the data structures and concepts (if not methods) in EWS while providing a truer RESTful experience to the consumer.

5.1 Data Providers

Three Data Providers are currently included in SmartConnector: [MongooseEwsRestProvider](#), [SoapEwsRestProvider](#), and [SboEwsRestProvider](#). As outlined [above](#), the names imply how each Data Provider obtains the resources to serve. It should be noted that not all Providers can provide all of the functionality which is defined in the API. This is expected and a by-product of the layered design approach implemented. This will be more apparent when reviewing the [Resources and Actions](#) section of this document.

5.1.1 MongooseEwsRestProvider

Serves data sourced from a native SmartConnector EWS Server.

5.1.2 SoapEwsRestProvider

Serves data sourced from any EWS Server using the standard EWS SOAP protocol. Any server which supports either the EWS v1.1 or EWS v1.2 specification can be used.

5.1.3 SboEwsRestProvider

Serves data for any SmartStruxure Building Operation (SBO) server using a combination of SOAP and native protocol commands.

5.2 Provisioning and Configuring Endpoints

Provisioning endpoints is performed via the SmartConnector portal. If you are not familiar with the SmartConnector Portal you should review section 5 of the *SmartConnector Installation and Configuration Guide*.

There number and type of REST endpoints you wish to provision in SmartConnector is only limited by the available resources on the host computer. Regardless of the type, the process is the same however some configuration parameters will differ depending on the type of Data Provider used.

5.2.1 Adding REST Endpoint

To provision a REST Endpoint in SmartConnector, perform the following:

1. Open SmartConnector Portal.
2. Select the Configurations-Endpoint menu option.
3. Click “Add New” from Endpoint Configurations page.
4. By default, only one assembly will be available so select the “Ews.RestExtensions” assembly from the list. If others are displayed consult the Extension Author’s documentation.

5. Click the “Next” button.
6. At this point you have an option to choose one of the Data Providers available. Select either Data Provider listed.
7. Click “Next”.
8. Edit the default Configuration Name and/or Description as desired.
9. Click “Finish”

SmartConnector will not immediately start the REST Endpoint because the configuration is not yet complete. If you click the “Validate” button, you will see what SmartConnector detects as issues. The actual issues will depend on which Data Provider you chose in step 6 above.

5.2.2 Configuring REST Endpoint

To complete the configuration, navigate to the “Endpoint Configuration” page for the REST Endpoint you wish to configure. If you are following from the prior section, you will be on this page after completing the “Add” steps above.

5.2.2.1 Common Configuration Parameters

The following parameters will be present regardless of the type of Data Provider you selected above. Those parameters which are specific to a Data Provider type are enumerated in the sections which follow.

When you click the “Details” tab, SmartConnector will display the details for the Data Provider you selected as shown in Figure 1.

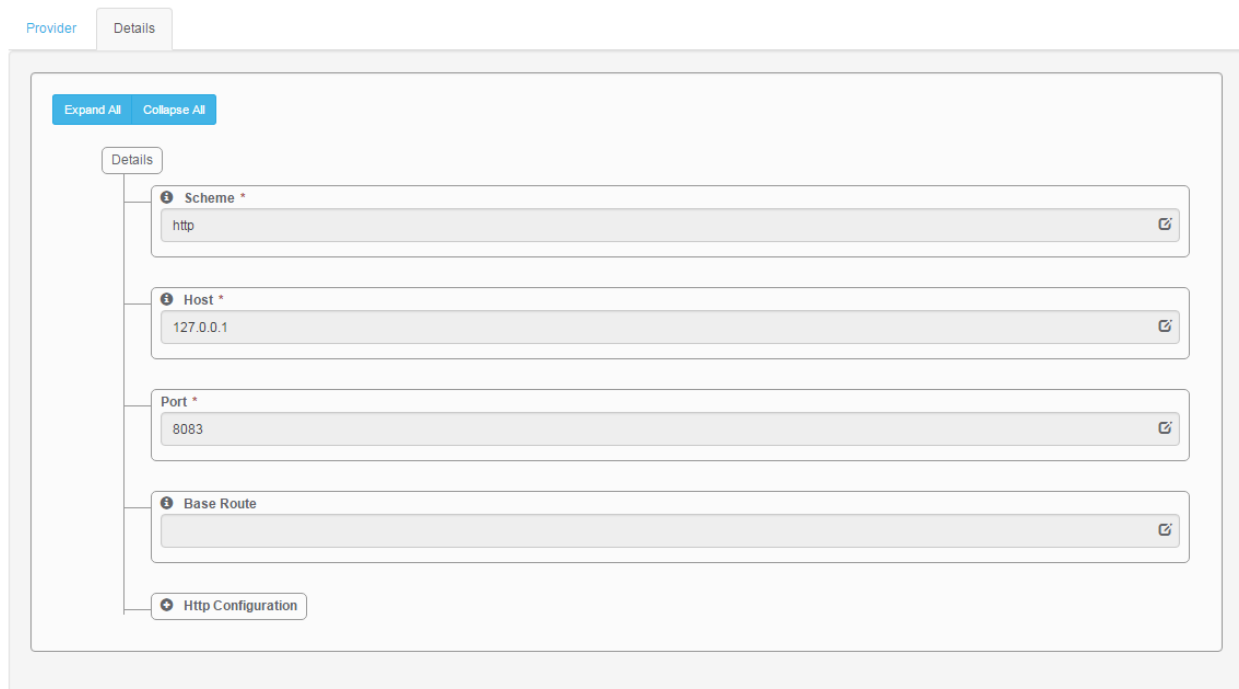


Figure 1: Portal - Configuration Details (Basic)

- **Scheme** – The scheme, HTTP or HTTPS, of the endpoint. See [Security Considerations](#) when deploying to a production environment.
- **Host** – The endpoint which is configured. This can be either an IP address or DNS name as desired.
- **Port** – The HTTP port used.
- **Base Route** – An optional base or prefix route which occurs before the standard routes.

Http Configuration Settings

Expanding the “Http Configuration” node, will display more options as shown in Figure 2.

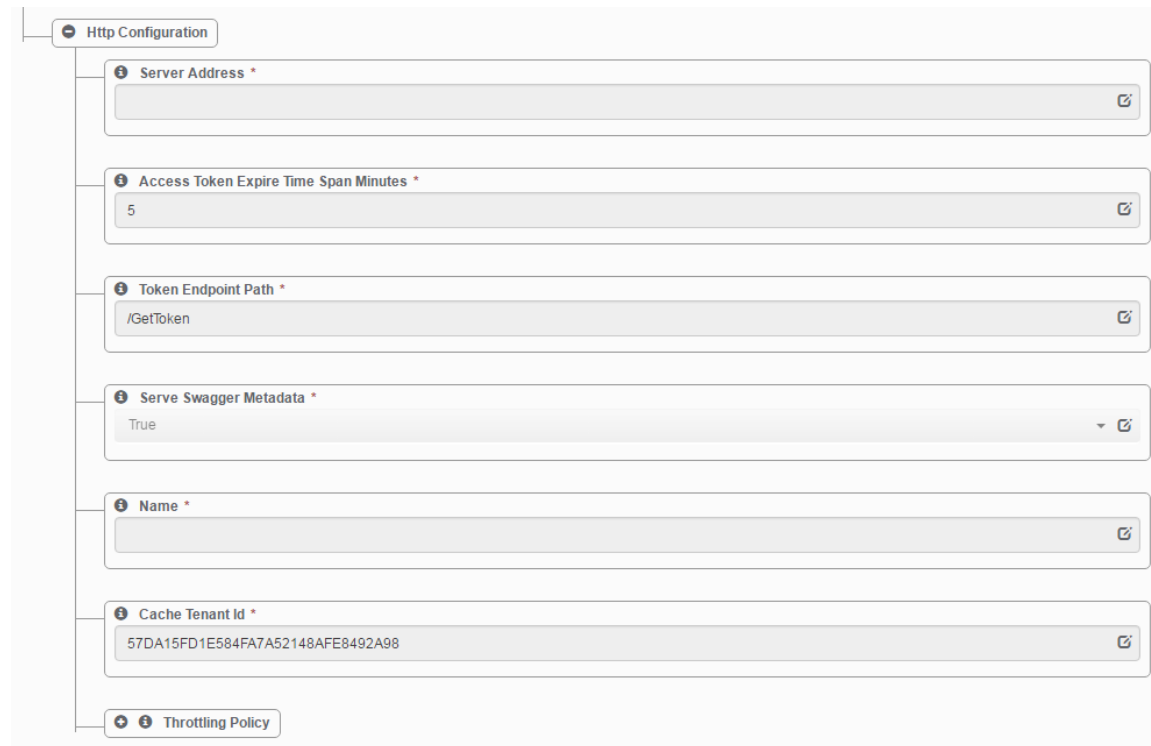
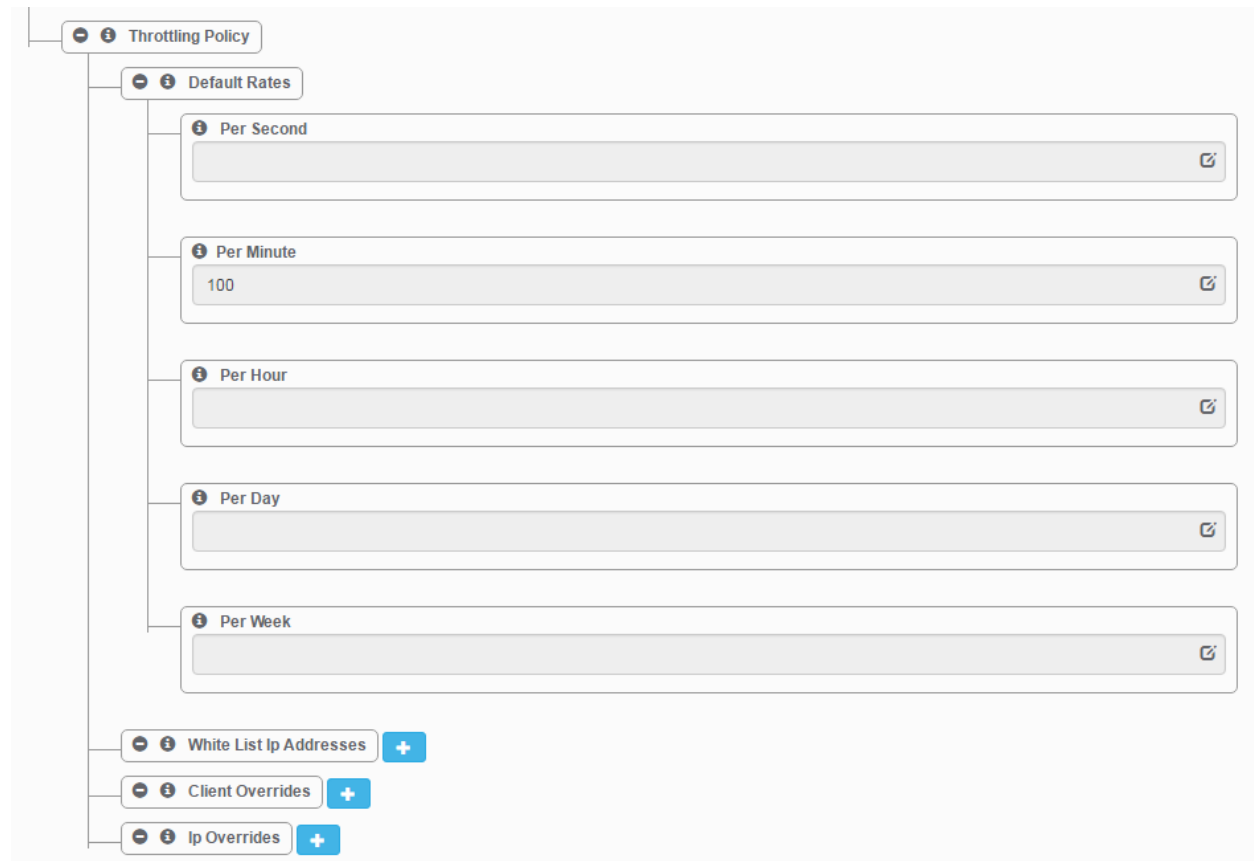


Figure 2: Portal - Configuration Details (HttpConfiguration)

- **Access Token Expire Time Span Minutes** – The amount of time the authorization Bearer Token is usable.
- **Token Endpoint Path** – The name of the route which is used to acquire the Bearer token.
- **Serve Swagger Metadata** – Whether or not Swagger and Swagger metadata are served.
- **Name** – How the Name of the endpoint will appear to Swagger users.
- **Cache Tenant Id** – Id of a SmartConnector InMemoryCache Tenant space allocated to the Data Provider.

Throttling Policy

Further expanding the “Throttling Policy” node, will display options for controlling call rates as shown in Figure 3.



The screenshot shows a configuration interface for a Throttling Policy. It is organized into a tree view with the following sections:

- Throttling Policy** (expanded)
 - Default Rates** (expanded)
 - Per Second: [Empty input field]
 - Per Minute: 100
 - Per Hour: [Empty input field]
 - Per Day: [Empty input field]
 - Per Week: [Empty input field]
 - White List Ip Addresses: [Add button]
 - Client Overrides: [Add button]
 - Ip Overrides: [Add button]

Figure 3: Portal - Configuration Details (Throttling Policy)

Default Rates

The “Default Rates” establish a base line for throttling. Typically, the values set here are the most restrictive but that is only a recommendation. Values work cooperatively where a “Per Second” limit and “Per Day” limit work in tandem. For example, calls may be limited to 2 per second but no more than 100 per day. An empty or “0” value indicates that no restriction is to be made.

- **Per Second** – The number of requests allowed per second.
- **Per Minute** – The number of requests allowed per minute.
- **Per Hour** – The number of requests allowed per hour.
- **Per Day** – The number of requests allowed per day.
- **Per Week** – The number of requests allowed per week.

White List Ip Addresses

Optional IP addresses, or range of addresses, can be added which would eliminate (white list) all calls made. This should only be used for trusted clients.

Client Overrides

Overrides to the base rates can be created based on an API Key. A unique key is assigned to a class of clients. These clients would then include the “key” in an “Authorization-Token” header in order to be governed by the updated rates.

- **Api Key** – Unique API key which will override the default rates with those set here.
- **Per Second** – The number of requests allowed per second.
- **Per Minute** – The number of requests allowed per minute.
- **Per Hour** – The number of requests allowed per hour.
- **Per Day** – The number of requests allowed per day.
- **Per Week** – The number of requests allowed per week.

Ip Overrides

Overrides to the base rates can be created based on a specific IP address.

- **Ip Address** – Ip address (or range) which will override the default rates with those set here.
- **Per Second** – The number of requests allowed per second.
- **Per Minute** – The number of requests allowed per minute.
- **Per Hour** – The number of requests allowed per hour.
- **Per Day** – The number of requests allowed per day.
- **Per Week** – The number of requests allowed per week.

5.2.2.2 *MongooseEwsRestProvider Configuration Parameters*

Http Configuration Settings

- **Ews Server Id** – The ID of the SmartConnector EWS Server which will be the data source. This value is displayed on the EWS Server Detail page. See *SmartConnector Installation and Configuration Guide* for more information.

5.2.2.3 *SoapEwsRestProvider Configuration Parameters*

Http Configuration Settings

- **Server Address** – The full SOAP endpoint for the EWS server which will be the data source.

5.2.2.4 *SboEwsRestProvider Configuration Parameters*

Http Configuration Settings

- **Server Domain** – The domain for the user credentials which will be used to connect to SBO.
- **Server Address** – The HTTP endpoint for the SBO server which will be used to serve data. This only needs to be the IP address (or DNS name) and port.
- **Search In Subservers** – Whether or not GET requests for lists objects will search in both the top level (e.g. and ES) server and children (e.g. ASes) or just the top level server.

5.2.3 Complete Configuration

Edit the configuration parameters described in the prior setting as needed. After you save your changes, you should be able to click the Validate button and see the “Endpoint Configuration valid” message. At that point, you will be able to click the “Start” button to complete provisioning of the REST endpoint. This endpoint will be started every time the SmartConnector service is started.

5.3 Resources and Actions

The resources served by RESTful EWS Gateway are derived from those found in the EWS specification. Some liberties were taken where appropriate. Actions on those resources strive to provide the full CRUD experience as would be expected in a RESTful API. It is important to note that not every Data Provider fully supports every action.

The following sections enumerate the resources and actions which comprise the RESTful EWS Gateway API. Each action additionally lists applicable parameters and their type. Data Provider restrictions are noted when appropriate.

5.3.1 AlarmEvents

An *AlarmEvent* represents the transition of an [Alarm](#) through the state transition model implemented. Every transition for an *Alarm*, including a return to normal, generates an *AlarmEvent* instance.

The properties of an *AlarmEvent* are listed in Table 1.

Property	Description	Data Type
Acknowledgeable	Acknowledge restrictions of the <i>AlarmEvent</i> .	<i>Enumeration</i> No=0 Yes= 1 Required=2 See section 4.6.2.1 from the EWS Specification for more details.
AlarmId	Optional Id of the associated Alarm .	<i>String</i>
Id	Case sensitive identifier for the <i>AlarmEvent</i> .	<i>String</i>
LastTransitionedOn	Date/Time in UTC when the transition captured by this <i>AlarmEvent</i> occurred.	<i>Date/Time</i>
Message	Human readable message for the <i>AlarmEvent</i> .	<i>String</i>
OccuredOn	Date/Time in UTC when the <i>Alarm</i> most recently transitioned into an “alarmed” state - that is not in the "Normal" state.	<i>Date/Time</i>
Priority	Priority value of the <i>AlarmEvent</i> .	<i>Integer</i>
State	State this <i>AlarmEvent</i> transition is documenting.	<i>Enumeration</i> Normal=0 Active=1 Acknowledged=2 Reset=3

		Disabled=4 See section 4.4.1 from the EWS Specification for more details.
Type	Server specific type of the <i>AlarmEvent</i> .	String

 Table 1: *AlarmEvent* Model

5.3.1.1 GET /AlarmEvents

Returns all *AlarmEvent* which fit the filter criteria supplied. All filter parameters are optional. *AlarmEvent* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
alarmId	Return only <i>AlarmEvent</i> for the referenced Alarm .	Query	String
priorityFrom	Return only <i>AlarmEvent</i> with a <i>Priority</i> greater or equal to the value supplied.	Query	Integer
priorityTo	Return only <i>AlarmEvent</i> with a <i>Priority</i> less or equal to the value supplied.	Query	Integer
state	Return only <i>AlarmEvent</i> with this <i>State</i> value.	Query	Enumeration (See Table 1)
acknowledgeable	Return only <i>AlarmEvent</i> with this <i>Acknowledgeable</i> value.	Query	Enumeration (See Table 1)
type	Return only <i>AlarmEvent</i> with this <i>Type</i> value.	Query	String
originallyAlarmedOnOrAfter	Return only <i>AlarmEvent</i> with <i>OccuredOn</i> which is on or after this value.	Query	DateTime
originallyAlarmedBefore	Return only <i>AlarmEvent</i> with <i>OccuredOn</i> which is before this value.	Query	DateTime
transitionedOnOrAfter	Return only <i>AlarmEvent</i> with <i>LastTransitionedOn</i> which is on or after this value.	Query	DateTime
transitionedBefore	Return only <i>AlarmEvent</i> with <i>LastTransitionedOn</i> which is before this value.	Query	DateTime
orderBy	<i>AlarmEvent</i> will be returned in this order. <i>TransitionedAtAscending</i> by default.	Query	Enumeration TransitionedAtAscending=0 TransitionedAtDescending=1 OriginallyAlarmedAtAscending=2 OriginallyAlarmedAtDescending=3

take	Number of <i>AlarmEvent</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>AlarmEvent</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 2: GET /AlarmEvents Parameters

5.3.1.2 GET /AlarmEvents/{id}

Returns the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEvent</i> requested. Must be double URL encoded.	Path	String

Table 3: GET /AlarmEvents/{id} Parameters

5.3.1.3 GET AlarmEvents/{id}/AlarmId

Returns the *AlarmId* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEvent</i> whose <i>AlarmId</i> property is to be returned. Must be double URL encoded.	Path	String

Table 4: GET /AlarmEvents/{id}/AlarmId Parameters

5.3.1.4 GET /AlarmEvents/{id}/Alarm

Returns the referenced [Alarm](#) for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEvent</i> for which the referenced <i>Alarm</i> is to be returned. Must be double URL encoded.	Path	String

Table 5: GET /AlarmEvents/{id}/Alarm Parameters

5.3.1.5 GET /AlarmEvents/{id}/Acknowledgeable

Returns the *Acknowledgeable* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEvent</i> whose <i>Acknowledgeable</i> property is to be returned. Must be double URL encoded.	Path	String

Table 6: GET /AlarmEvents/{id}/Acknowledgeable Parameters

5.3.1.6 GET /AlarmEvents/{id}/OccurredOn

Returns the *OccurredOn* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>AlarmEvent</i> whose <i>OccurredOn</i> property is to be returned. Must be double URL encoded.	Path	String

Table 7: GET /AlarmEvents/{id}/OccurredOn Parameters

5.3.1.7 GET /AlarmEvents/{id}/LastTransitionedOn

Returns the *LastTransitionedOn* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>AlarmEvent</i> whose <i>LastTransitionedOn</i> property is to be returned. Must be double URL encoded.	Path	String

Table 8: GET /AlarmEvents/{id}/LastTransitionedOn Parameters

5.3.1.8 GET /AlarmEvents/{id}/Message

Returns the *Message* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>AlarmEvent</i> whose <i>Message</i> property is to be returned. Must be double URL encoded.	Path	String

Table 9: GET /AlarmEvents/{id}/Message Parameters

5.3.1.9 GET /AlarmEvents/{id}/Priority

Returns the *Priority* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>AlarmEvent</i> whose <i>Priority</i> property is to be returned. Must be double URL encoded.	Path	String

Table 10: GET /AlarmEvents/{id}/Priority Parameters

5.3.1.10 GET /AlarmEvents/{id}/State

Returns the *State* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>AlarmEvent</i> whose <i>State</i> property is to be returned. Must be double URL encoded.	Path	String

Table 11: GET /AlarmEvents/{id}/State Parameters

5.3.1.11 GET /AlarmEvents/{id}/Type

Returns the *Type* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>AlarmEvent</i> whose <i>Type</i> property is to be returned. Must be double URL encoded.	Path	String

Table 12: GET /AlarmEvents/{id}/Type Parameters

5.3.1.12 POST /AlarmEvents/{id}/Acknowledge

Executes an Acknowledge operation on the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>AlarmEvent</i> to Acknowledge. Must be double URL encoded.	Path	String

Table 13: GET /AlarmEvents/{id}/Acknowledge Parameters

5.3.2 AlarmEventTypes

An *AlarmEventType* represents a server specific a Type for an [AlarmEvent](#).

The properties of an *AlarmEventType* are listed in Table 14.

Property	Description	Data Type
Description	Optional description of the <i>AlarmEventType</i> .	String
Id	Case sensitive identifier for the <i>AlarmEventType</i> .	String
Name	Name of the <i>AlarmEventType</i> .	String

Table 14: AlarmEventType Model

5.3.2.1 POST /AlarmEventTypes/Create

Creates a new *AlarmEventType* and returns it.

Parameter	Description	Parameter Type	Data Type
-----------	-------------	----------------	-----------

newItem	Definition of how the <i>AlarmEventType</i> will be created.	<i>Body</i>	<i>Object</i>
Description	Description given to the new <i>AlarmEventType</i> .		<i>String</i>
Name	Name given to the new <i>AlarmEventType</i> ,		<i>String</i>

Table 15: POST /AlarmEventTypes/Create Parameters

SoapEwsRestProvider does not support this action.

5.3.2.2 GET /AlarmEventTypes

Returns all *AlarmEventType* which fit the filter criteria supplied. All filter parameters are optional. *AlarmEventType* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
name	Return only <i>AlarmEventType</i> whose <i>Name</i> contains this value.	<i>Query</i>	<i>String</i>
orderBy	<i>AlarmEventType</i> will be returned in this order. <i>NameAscending</i> by default.	<i>Query</i>	<i>Enumeration</i> NameAscending=0 NameDescending=1
take	Number of <i>AlarmEventType</i> that should be returned. 200 by default.	<i>Query</i>	<i>Integer</i>
skip	Number of <i>AlarmEventType</i> that should be skipped before items are returned. 0 by default.	<i>Query</i>	<i>Integer</i>

Table 16: GET /AlarmEventTypes Parameters

5.3.2.3 GET /AlarmEventTypes/{id}

Returns the *AlarmEventType* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEventType</i> requested. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 17: GET /AlarmEventTypes/{id} Parameters

5.3.2.4 GET /AlarmEventTypes/{id}/Description

Returns the *Description* property for the *AlarmEventType* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEventType</i> whose <i>Description</i> property is to be	<i>Path</i>	<i>String</i>

	returned. Must be double URL encoded.		
--	---------------------------------------	--	--

Table 18: GET /AlarmEventTypes/{id}/Description Parameters

5.3.2.5 PUT /AlarmEventTypes/{id}/Description

Modifies the *Description* property for the *AlarmEventType* with the *Id* provided to the new value supplied. Returns the modified *AlarmEventType*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEventType</i> whose <i>Description</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Description</i> value.	<i>Body</i>	<i>String</i>

Table 19: PUT /AlarmEventTypes/{id}/Description Parameters

SoapEwsRestProvider does not support this action.

5.3.2.6 GET /AlarmEventTypes/{id}/Name

Returns the *Name* property for the *AlarmEventType* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEventType</i> whose <i>Name</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 20: GET /AlarmEventTypes/{id}/Name Parameters

5.3.2.7 PUT /AlarmEventTypes/{id}/Name

Modifies the *Name* property for the *AlarmEventType* with the *Id* provided to the new value supplied. Returns the modified *AlarmEventType*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEventType</i> whose <i>Name</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Name</i> value.	<i>Body</i>	<i>String</i>

Table 21: PUT /AlarmEventTypes/{id}/Name Parameters

SoapEwsRestProvider does not support this action.

5.3.2.8 PATCH /AlarmEventTypes/{id}/Update

Modifies the *AlarmEventType* with the *Id* provided to the values supplied. Returns the modified *AlarmEventType*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEventType</i> to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	Values to update the <i>AlarmEventType</i> with.	<i>Body</i>	<i>Object</i>
Description	New <i>Description</i> value.		<i>String</i>
Name	New <i>Name</i> value.		<i>String</i>

Table 22: PATCH /AlarmEventTypes/{id}/Update Parameters

SoapEwsRestProvider does not support this action.

5.3.2.9 DELETE /AlarmEventTypes/{id}/Delete

Deletes the *AlarmEventType* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>AlarmEventType</i> to be deleted. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 23: DELETE /AlarmEventTypes/{id}/Delete Parameters

SoapEwsRestProvider does not support this action.

5.3.3 Alarms

The existence of an *Alarm* does not imply that an alarm has occurred, only that alarm criteria exists. Typically, this is done for a specific *Value* but this is not required. If *Alarm* is associated with a *Value*, the *Id* of that [Value](#) should be included.

The properties of an *Alarm* are listed in Table 24.

Property	Description	Data Type
Description	Optional description of the <i>Alarm</i> .	<i>String</i>
Id	Case sensitive identifier for the <i>Alarm</i> .	<i>String</i>
Name	<i>Name</i> of the <i>Alarm</i> .	<i>String</i>
ParentId	<i>Id</i> of the Container which the <i>Alarm</i> is found in.	<i>String</i>

State	Current state of the <i>Alarm</i>	<i>Enumeration</i> Normal=1 Active=1 Acknowledged=2 Reset=3 Disabled=4 See Table 11 from the EWS Specification for more details.
TransitionModel	The transition model for the <i>Alarm</i> .	<i>Enumeration</i> Unrestricted=0 SimpleSystemAlarm=1 NoAcknowledgeRequired=2 SimpleTransientAlarm=3 SingleAcknowledgeRequirement=4 ExtendedAcknowledgeRequirement=5 See section 4.6.1 from the EWS Specification for more details.
ValueId	Optional <i>Id</i> of the <i>Value</i> the <i>Alarm</i> is associated with.	<i>String</i>

Table 24: Alarm Model

5.3.3.1 POST /Alarms/Create

Creates a new *Alarm* and returns it.

Parameter	Description	Parameter Type	Data Type
newItem	Definition of how the <i>Alarm</i> will be created.	<i>Body</i>	<i>Object</i>
Description	<i>Description</i> for the new <i>Alarm</i> .		<i>String</i>
Id	<i>Id</i> for the new <i>Alarm</i> .		<i>String</i>
Name	<i>Name</i> for the new <i>Alarm</i> .		<i>String</i>
ParentId	Optional <i>Id</i> of the Container the new <i>Alarm</i> will be put into.		<i>String</i>
TransitionModel	<i>TransitionModel</i> for the new <i>Alarm</i> .		<i>Enumeration</i> See Table 24.
ValueId	Optional <i>Id</i> of the Value the new <i>Alarm</i> will be associated with.		<i>String</i>

Table 25: POST /Alarms/Create Parameters

SoapEwsRestProvider does not support this action.

5.3.3.2 GET /Alarms

Returns all *Alarm* which fit the filter criteria supplied. All filter parameters are optional. *Alarm* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
name	Return only <i>Alarm</i> whose name contains this value.	Query	String
state	Return only <i>Alarm</i> with this <i>State</i> value.	Query	Enumeration See Table 24
orderBy	<i>Alarm</i> will be returned in this order. <i>NameAscending</i> by default.	Query	Enumeration NameAscending=0 NameDescending=1
take	Number of <i>Alarm</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>Alarm</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 26: GET /Alarms Parameters

SoapEwsRestProvider does not support this action.

5.3.3.3 GET /Alarms/{id}

Returns the *Alarm* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	Id of the <i>Alarm</i> requested. Must be double URL encoded.	Path	String

Table 27: GET /Alarms/{id} Parameters

5.3.3.4 GET /Alarms/{id}/Description

Returns the *Description* property for the *Alarm* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> whose <i>Description</i> property is to be returned. Must be double URL encoded.	Path	String

Table 28: GET /Alarms/{id}/Description Parameters

5.3.3.5 PUT /Alarms/{id}/Description

Modifies the *Description* property for the *Alarm* with the *Id* provided to the new value supplied. Returns the modified *Alarm*.

Parameter	Description	Parameter Type	Data Type
-----------	-------------	----------------	-----------

id	<i>Id</i> of the <i>Alarm</i> whose <i>Description</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Description</i> value.	<i>Body</i>	<i>String</i>

Table 29: PUT /Alarms/{id}/Description Parameters

SoapEwsRestProvider does not support this action.

5.3.3.6 GET /Alarms/{id}/Name

Returns the *Name* property for the *Alarm* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> whose <i>Name</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 30: GET /Alarms/{id}/Name Parameters

5.3.3.7 PUT /Alarms/{id}/Name

Modifies the *Name* property for the *Alarm* with the *Id* provided to the new value supplied. Returns the modified *Alarm*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> whose <i>Name</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Name</i> value.	<i>Body</i>	<i>String</i>

Table 31: PUT /Alarms/{id}/Name Parameters

SoapEwsRestProvider does not support this action.

5.3.3.8 GET /Alarms/{id}/Parent

Returns the parent [Container](#) for the *Alarm* with the *Id* supplied.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> whose parent <i>Container</i> is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 32: GET /Alarms/{id}/Parent Parameters

5.3.3.9 GET /Alarms/{id}/ParentId

Returns the *ParentId* property for the *Alarm* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the Alarm whose ParentId property is to be returned. Must be double URL encoded.	Path	String

Table 33: GET /Alarms/{id}/ParentId Parameters

5.3.3.10 PUT /Alarms/{id}/ParentId

Modifies the ParentId property for the Alarm with the Id provided to the new value supplied. Returns the modified Alarm.

Parameter	Description	Parameter Type	Data Type
id	Id of the Alarm whose ParentId property is to be modified. Must be double URL encoded.	Path	String
newValue	New ParentId value.	Body	String

Table 34: PUT /Alarms/{id}/ParentId Parameters

SoapEwsRestProvider does not support this action.

5.3.3.11 GET /Alarms/{id}/State

Returns the State property for the Alarm with the Id provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the Alarm whose State property is to be returned. Must be double URL encoded.	Path	String

Table 35: GET /Alarms/{id}/State Parameters

5.3.3.12 GET /Alarms/{id}/TransitionModel

Returns the TransitionModel property for the Alarm with the Id provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the Alarm whose TransitionModel property is to be returned. Must be double URL encoded.	Path	String

Table 36: GET /Alarms/{id}/TransitionModel Parameters

SoapEwsRestProvider does support this action however, the value returned will always be *Unrestricted* because the EWS SOAP protocol does not directly support this property even though it discusses the topic of state transition at length.

5.3.3.13 GET /Alarms/{id}/Value

Returns the associated [Value](#) for the *Alarm* with the *Id* supplied.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> whose associated <i>Value</i> is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 37: GET /Alarms/{id}/Value Parameters

5.3.3.14 GET /Alarms/{id}/ValueId

Returns the *ValueId* property for the *Alarm* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> whose <i>ValueId</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 38: GET /Alarms/{id}/ValueId Parameters

5.3.3.15 PATCH /Alarms/{id}/Update

Modifies the *Alarm* with the *Id* provided to the values supplied. Returns the modified *Alarm*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
updatedItem	Values to update the <i>Alarm</i> with.	<i>Body</i>	<i>Object</i>
Description	New <i>Description</i> value.		<i>String</i>
Name	New <i>Name</i> value.		<i>String</i>
ParentId	New <i>ParentId</i> value.		<i>String</i>

Table 39: PATCH /Alarm/{id}/Update Parameters

SoapEwsRestProvider does not support this action.

5.3.3.16 PUT /Alarms/{id}/Id

Modifies the *Id* property for the *Alarm* with the *Id* provided to the new value supplied. Returns the modified *Alarm*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> whose <i>Id</i> property is to be modified. Must be double	<i>Path</i>	<i>String</i>

	URL encoded.		
newValue	New <i>Id</i> value.	<i>Body</i>	<i>String</i>

Table 40: PUT /Alarms/{id}/Id Parameters

SoapEwsRestProvider does not support this action.

5.3.3.17 POST /Alarms/{id}/Acknowledge

Executes an Acknowledge action on the *Alarm* with the *Id* provided. The result will generate a new [AlarmEvent](#) based on the information contained in the newEvent parameter. Restrictions apply based on the *TransitionModel* of the *Alarm*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> to be acknowledged. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newEvent	Information to describe the generated <i>AlarmEvent</i> .	<i>Body</i>	<i>Object</i>
Message	Message for the generated <i>AlarmEvent</i> .		<i>String</i>
Priority	Priority for the generated <i>AlarmEvent</i> .		<i>Integer</i>
Type	Server specific type for the generated <i>AlarmEvent</i> .		<i>String</i>

Table 41: POST /Alarm/{id}/Acknowledge Parameters

SoapEwsRestProvider does not support this action.

5.3.3.18 POST /Alarms/{id}/Activate

Executes an Activate action on the *Alarm* with the *Id* provided. The result will generate a new [AlarmEvent](#) based on the information contained in the newEvent parameter. Restrictions apply based on the *TransitionModel* of the *Alarm*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> to be activated. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newEvent	Information to describe the generated <i>AlarmEvent</i> .	<i>Body</i>	<i>Object</i>
Message	<i>Message</i> for the generated <i>AlarmEvent</i> .		<i>String</i>
Priority	<i>Priority</i> for the generated <i>AlarmEvent</i> .		<i>Integer</i>

Type	Server specific type for the generated <i>AlarmEvent</i> .		<i>String</i>
------	--	--	---------------

Table 42: POST /Alarm/{id}/Activate Parameters

SoapEwsRestProvider does not support this action.

5.3.3.19 POST /Alarms/{id}/Disable

Executes a Disable action on the *Alarm* with the *Id* provided. The result will generate a new [AlarmEvent](#) based on the information contained in the *newEvent* parameter. Restrictions apply based on the *TransitionModel* of the *Alarm*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> to be disabled. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newEvent	Information to describe the generated AlarmEvent .	<i>Body</i>	<i>Object</i>
Message	Message for the generated <i>AlarmEvent</i> .		<i>String</i>
Priority	Priority for the generated <i>AlarmEvent</i> .		<i>Integer</i>
Type	Server specific type for the generated <i>AlarmEvent</i> .		<i>String</i>

Table 43: POST /Alarm/{id}/Disable Parameters

SoapEwsRestProvider does not support this action.

5.3.3.20 POST /Alarms/{id}/Reset

Executes a Reset action on the *Alarm* with the *Id* provided. The result will generate a new [AlarmEvent](#) based on the information contained in the *newEvent* parameter. Restrictions apply based on the *TransitionModel* of the *Alarm*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> to be reset. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newEvent	Information to describe the generated <i>AlarmEvent</i> .	<i>Body</i>	<i>Object</i>
Message	Message for the generated <i>AlarmEvent</i> .		<i>String</i>
Priority	Priority for the generated <i>AlarmEvent</i> .		<i>Integer</i>

Type	Server specific type for the generated <i>AlarmEvent</i> .		String
------	--	--	--------

Table 44: POST /Alarm/{id}/Reset Parameters

SoapEwsRestProvider does not support this action.

5.3.3.21 POST /Alarms/{id}/ChangeState

Executes a manual state change action on the *Alarm* with the *Id* provided (*Alarm* must have an Unrestricted *TransitionModel*). The result will generate a new [AlarmEvent](#) based on the information contained in the *newEvent* parameter.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> to be reset. Must be double URL encoded.	Path	String
newEvent	Information to describe the generated <i>AlarmEvent</i> .	Body	Object
Message	Message for the generated <i>AlarmEvent</i> .		String
Priority	Priority for the generated <i>AlarmEvent</i> .		Integer
Type	Server specific type for the generated <i>AlarmEvent</i> .		String

Table 45: POST /Alarm/{id}/ChangeState Parameters

SoapEwsRestProvider does not support this action.

5.3.3.22 DELETE /Alarms/{id}/Delete

Deletes the *Alarm* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Alarm</i> to be deleted. Must be double URL encoded.	Path	String

Table 46: DELETE /Alarms/{id}/Delete Parameters

SoapEwsRestProvider does not support this action.

5.3.4 Containers

A *Container* is the parent for other types; namely [Value](#), [Alarm](#), [Container](#), and [Trend](#).

Property	Description	Data Type
Description	Optional description of the <i>Container</i> .	String

Id	Case sensitive identifier for the <i>Container</i> .	String
Name	Name of the <i>Container</i> .	String
ParentId	<i>Id</i> of the <i>Container</i> which the <i>Container</i> is found in.	String
Type	Type of the <i>Container</i> .	Enumeration Folder=0 Server=1 Device=2 Structure=3 Service=4 See Section 4.2.1 from the EWS Specification for more details.

Table 47: Container Model

5.3.4.1 POST /Containers/Create

Creates a new *Container* and returns it.

Parameter	Description	Parameter Type	Data Type
newItem	Definition of how the <i>Container</i> will be created.	Body	Object
Description	<i>Description</i> for the new <i>Container</i> .		String
Id	<i>Id</i> for the new <i>Container</i> .		String
Name	<i>Name</i> for the new <i>Container</i> .		String
ParentId	Optional <i>Id</i> of the <i>Container</i> the new <i>Container</i> will be put into.		String
Type	<i>Type</i> for the new <i>Container</i> .		Enumeration See Table 47

Table 48: POST /Containers/Create Parameters

SoapEwsRestProvider does not support this action.

5.3.4.2 GET /Containers

Returns all *Container* which fit the filter criteria supplied. All filter parameters are optional. *Container* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
name	Return only <i>Container</i> whose <i>Name</i> contains this value.	Query	String
type	Return only <i>Container</i> with this <i>Type</i> value.	Query	Enumeration See Table 47

orderBy	Container will be returned in this order. <i>NameAscending</i> by default.	Query	Enumeration NameAscending=0 NameDescending=1
take	Number of <i>Container</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>Container</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 49: GET /Containers Parameters

SoapEwsRestProvider does not support this action.

5.3.4.3 GET /Containers/{id}

Returns the *Container* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> requested. Must be double URL encoded.	Path	String

Table 50: GET /Containers/{id} Parameters

SoapEwsRestProvider does not support this action.

5.3.4.4 GET /Containers/{id}/Children

Returns all children ([Value](#), [Alarm](#), [Trend](#) or *Container*) parented by the *Container* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose children are to be returned. Must be double URL encoded.	Path	String
type	Return only those children which are of this type. All returned by default.	Query	Enumeration Container=0 Value=1 Trend=2 Alarm=3

Table 51: GET /Containers/{id}/Children Parameters

5.3.4.5 GET /Containers/{id}/Description

Returns the *Description* property for the *Container* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose	Path	String

	<i>Description</i> property is to be returned. Must be double URL encoded.		
--	--	--	--

Table 52: GET /Containers/{id}/Description Parameters

5.3.4.6 PUT /Containers/{id}/Description

Modifies the *Description* property for the *Container* with the *Id* provided to the new value supplied. Returns the modified *Container*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>Description</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Description</i> value.	<i>Body</i>	<i>String</i>

Table 53: PUT /Containers/{id}/Description Parameters

SoapEwsRestProvider does not support this action.

5.3.4.7 GET /Containers/{id}/Name

Returns the *Name* property for the *Container* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>Name</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 54: GET /Containers/{id}/Name Parameters

5.3.4.8 PUT /Containers/{id}/Name

Modifies the *Name* property for the *Container* with the *Id* provided to the new value supplied. Returns the modified *Container*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>Name</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Name</i> value.	<i>Body</i>	<i>String</i>

Table 55: PUT /Containers/{id}/Name Parameters

SoapEwsRestProvider does not support this action.

5.3.4.9 GET /Containers/{id}/Parent

Returns the parent [Container](#) for the *Container* with the *Id* supplied.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose parent <i>Container</i> is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 56: GET /Containers/{id}/Parent Parameters

5.3.4.10 GET /Containers/{id}/ParentId

Returns the *ParentId* property for the *Container* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>ParentId</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 57: GET /Containers/{id}/ParentId Parameters

5.3.4.11 PUT /Containers/{id}/ParentId

Modifies the *ParentId* property for the *Container* with the *Id* provided to the new value supplied.

Returns the modified *Container*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>ParentId</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>ParentId</i> value.	<i>Body</i>	<i>String</i>

Table 58: PUT /Containers/{id}/ParentId Parameters

SoapEwsRestProvider does not support this action.

5.3.4.12 GET /Containers/{id}/Type

Returns the *Type* property for the *Container* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>Type</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 59: GET /Containers/{id}/Type Parameters

5.3.4.13 PUT /Containers/{id}/Type

Modifies the *Type* property for the *Container* with the *Id* provided to the new value supplied. Returns the modified *Container*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>Type</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Type</i> value.	<i>Body</i>	<i>Enumeration</i> See Table 47

Table 60: PUT /Containers/{id}/Type Parameters

SoapEwsRestProvider does not support this action.

5.3.4.14 PATCH /Containers/{id}/Update

Modifies the *Container* with the *Id* provided to the values supplied. Returns the modified *Container*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
updatedItem	Values to update the <i>Container</i> with.	<i>Body</i>	<i>Object</i>
Description	New <i>Description</i> value.		<i>String</i>
Id	New <i>Id</i> value.		<i>String</i>
Name	New <i>Name</i> value.		<i>String</i>
ParentId	New <i>ParentId</i> value.		<i>String</i>
Type	New <i>Type</i> value.		<i>Enumeration</i> See Table 47.

Table 61: PATCH /Containers/{id}/Update Parameters

SoapEwsRestProvider does not support this action.

5.3.4.15 PUT /Containers/{id}/Id

Modifies the *Id* property for the *Container* with the *Id* provided to the new value supplied. Returns the modified *Container*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> whose <i>Id</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>

newValue	New <i>Id</i> value.	Body	String
----------	----------------------	------	--------

Table 62: PUT /Containers/{id}/Id Parameters

SoapEwsRestProvider does not support this action.

5.3.4.16 DELETE /Containers/{id}/Delete

Deletes the *Container* with the *Id* provided. All children will also be deleted.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Container</i> to be deleted. Must be double URL encoded.	Path	String

Table 63: DELETE /Containers/{id}/Delete Parameters

SoapEwsRestProvider does not support this action.

5.3.5 Notifications

A *Notification* represents a logical session of [NotificationItem](#) for a [Subscription](#) at a point in time.

Property	Description	Data Type
CreatedOn	Date/Time in UTC when the <i>Notification</i> session was created.	Date/Time
Id	Case sensitive identifier for the <i>Value</i> .	String
SubscriptionId	The <i>Subscription</i> that this <i>Notification</i> is for.	String

Table 64: Notification Model

5.3.5.1 POST /Notifications/Create

Creates a new *Notification* and returns it.

Parameter	Description	Parameter Type	Data Type
newItem	Definition of how the <i>Notification</i> will be created.	Body	Object
SubscriptionId	<i>Id</i> of the <i>Subscription</i> the <i>Notification</i> will be created under.		String
ChangesOnly	If true, all items will be returned. If false, only changes since the last <i>Notification</i> for the <i>Subscription</i> will be created.		Boolean

Table 65: POST /Notifications/Create Parameters

5.3.5.2 GET /Notifications

Returns all *Notification* which fit the filter criteria supplied. All filter parameters are optional. *Notification* will be ordered and pagged as requested.

Parameter	Description	Parameter Type	Data Type
subscriptionId	Return only <i>Notification</i> with this <i>SubscriptionId</i> value.	Query	String
orderBy	<i>Notification</i> will be returned in this order. <i>CreatedOnAscending</i> by default.	Query	Enumeration CreatedOnAscending=0 CreatedOnDescending=1
take	Number of <i>Notification</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>Notification</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 66: GET /Notifications Parameters

5.3.5.3 GET /Notifications/{id}

Returns the *Notification* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Notification</i> requested. Must be double URL encoded.	Path	String

Table 67: GET /Notifications/{id} Parameters

5.3.5.4 GET /Notifications/{id}/CreatedOn

Returns the *CreatedOn* property for the *Notification* with the *Id* provided

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Notification</i> whose <i>CreatedOn</i> property is to be returned. Must be double URL encoded.	Path	String

Table 68: GET /Notifications/{id}/CreatedOn Parameters

5.3.5.5 GET /Notifications/{id}/Items

Returns *NotificationItem* for the *Notification* with the *Id* provided. Sort order for the returned items depends on *Subscription.Type*. For *ValueItemChanged* and *HierarchyChanged*, the sort order is by *ChangedAt*. For *AlarmEventChanged*, the sort order is by *ChangedItemId*.

A *NotificationItem* represents a single event notification for given *Notification* session belonging to a given [Subscription](#). The properties of a *NotificationItem* are listed in Table 69.

Property	Data Type	Description
----------	-----------	-------------

Id	<i>String</i>	Case sensitive identifier for the <i>NotificationItem</i> .
SubscriptionId	<i>String</i>	The Id of the Subscription that this <i>NotificationItem</i> is for.
NotificationId	<i>String</i>	The Id of the <i>Notification</i> that this <i>NotificationItem</i> is for.
ChangeType	<i>Enumeration</i> ValueItemChanged=0 AlarmEventChanged=1 SystemEventChanged=2 HierarchyChanged=3 See Section 4.11.2 from the EWS Specification for more details.	The type of change this is a <i>NotificationItem</i> for.
ChangedItemId	<i>String</i>	Id of the item that changed.
ChangedItemType	<i>Enumeration</i> Container=0, Value=1 Trend=2 Alarm=3	The type of entity which <i>ChangedItemId</i> refers to.
ChangedAt	<i>DateTime</i>	The <i>DateTime</i> in UTC when the change occurred. Not available for all <i>ChangeType</i> values.
Value	<i>String</i>	The Value at the time of the change. (<i>ChangeType</i> = <i>ValueItemChanged</i> only).
ValueState	<i>Enumeration</i> Good=0 Uncertain=1 Forced=2 Offline=3 Error=4 See section 4.3.4 of the EWS Specification for more details.	The State at the time of the change. (<i>ChangeType</i> = <i>ValueItemChanged</i> only).
AlarmEventState	<i>Enumeration</i>	The Alarm State at the time of the change. (<i>ChangeType</i> = <i>AlarmEventChanged</i> only).

Table 69: NotificationItem Model

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Notification</i> whose items are to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>
take	Number of <i>NotificationItems</i> that	<i>Query</i>	<i>Integer</i>

	should be returned. 200 by default.		
skip	Number of <i>NotificationItems</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 70: GET /Notifications/{id}/Items Parameters

5.3.5.6 GET /Notifications/{id}/Subscription

Returns the *Subscription* for the *Notification* with the *Id* supplied.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Notification</i> whose Subscription is to be returned. Must be double URL encoded.	Path	String

Table 71: GET /Notifications/{id}/Subscription Parameters

5.3.5.7 GET /Notifications/{id}/SubscriptionId

Returns the *SubscriptionId* property for the *Notification* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Notification</i> whose <i>SubscriptionId</i> property is to be returned. Must be double URL encoded.	Path	String

Table 72: GET /Notifications/{id}/SubscriptionId Parameters

5.3.5.8 DELETE /Notifications/{id}/Delete

Deletes the *Notification* with the *Id* provided. All *NotificationItems* will also be deleted.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Notification</i> to be deleted. Must be double URL encoded.	Path	String

Table 73: DELETE /Notifications/{id}/Delete Parameters

5.3.6 Root

Since it is not always feasible to know the *Id* of the “root” *Container*, a special resource named “Root” provides that *Container*.

5.3.6.1 GET /

Returns the root [Container](#) which all other data is rooted.

5.3.7 Subscriptions

A Subscription is as data registration to one or more items for the purpose of creating [Notification](#) sessions to retrieve [NotificationItem](#) when the subscribed items have changed.

Property	Description	Data Type
Id	Case sensitive identifier for the <i>Subscription</i> .	<i>String</i>
Type	The type of change this is a <i>Subscription</i> for.	<i>Enumeration</i> ValueItemChanged=0 AlarmEventChanged=1 SystemEventChanged=2 HierarchyChanged=3 See Section 4.11.2 from the EWS Specification for more details.
IsActive	Indicates if the Subscription is active and new Notification can be generated	<i>Boolean</i>
ExpiresOn	DateTime in UTC when the Subscription will expire.	<i>DateTime</i>

Table 74: Subscription Model

5.3.7.1 POST /Subscriptions/Create

Creates a new *Subscription* and returns it.

Parameter	Description	Parameter Type	Data Type
newItem	Definition of how the <i>Subscription</i> will be created.	<i>Body</i>	<i>Object</i>
DurationInMinutes	The requested duration for the <i>Subscription</i> .		<i>Integer</i>
SubscriptionType	The type of data to subscribe to.		<i>Enumeration</i> See Table 74
Ids	List of Ids of items which are being Subscribed to.		<i>String[]</i>
PriorityFrom	For <i>AlarmEventChanged</i> and <i>SystemEventChanged</i> subscriptions, only items with Priority at or above this value will be included.		<i>Integer</i>
PriorityTo	For <i>AlarmEventChanged</i> and <i>SystemEventChanged</i> subscriptions, only items with Priority below this value will be included.		<i>Integer</i>
Types	For <i>AlarmEventChanged</i> and <i>SystemEventChanged</i> subscriptions, only items with a Type in this List will be included.		<i>String[]</i>

Table 75: POST /Subscriptions/Create Parameters

5.3.7.2 GET /Subscriptions

Returns all *Subscription* which fit the filter criteria supplied. All filter parameters are optional. *Subscription* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
type	Return only <i>Subscription</i> with this <i>Type</i> value.	Query	Enumeration See Table 74
orderBy	<i>Subscription</i> will be returned in this order. <i>ExpiresAscending</i> by default.	Query	Enumeration <i>ExpiresAscending</i> =0 <i>ExpiresDescending</i> =1
take	Number of <i>Subscription</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>Subscription</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 76: GET /Subscriptions Parameters

5.3.7.3 GET /Subscriptions/{id}

Returns the *Subscription* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Subscription</i> requested. Must be double URL encoded.	Path	String

Table 77: GET /Subscriptions/{id} Parameters

5.3.7.4 GET /Subscriptions/{id}/ExpiresOn

Returns the *ExpiresOn* property for the *Subscription* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Subscription</i> whose <i>ExpiresOn</i> property is to be returned. Must be double URL encoded.	Path	String

Table 78: GET /Subscriptions/{id}/ExpiresOn Parameters

5.3.7.5 GET /Subscriptions/{id}/Type

Returns the *Type* property for the *Subscription* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
-----------	-------------	----------------	-----------

id	<i>Id</i> of the <i>Subscription</i> whose <i>Type</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>
----	--	-------------	---------------

Table 79: GET /Subscriptions/{id}/Type Parameters

5.3.7.6 GET /Subscriptions/{id}/IsActive

Returns the *IsActive* property for the *Subscription* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Subscription</i> whose <i>IsActive</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 80: GET /Subscriptions/{id}/IsActive Parameters

5.3.7.7 PUT /Subscriptions/{id}/Renew

Renews the *Subscription* with the *Id* provided by the requested number of minutes. Returns the renewed *Subscription*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Subscription</i> to be renewed. Must be double URL encoded.	<i>Path</i>	<i>String</i>
Minutes	Value in minutes to extend the <i>Subscription</i> .	<i>Body</i>	<i>Integer</i>

Table 81: PUT /Subscriptions/{id}/Renew Parameters

5.3.7.8 PUT /Subscriptions/{id}/Terminate

Terminates the *Subscription* with the *Id*. Terminated *Subscription* no longer can be used to create [Notification](#) sessions. Returns the terminated *Subscription*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Subscription</i> to be terminated. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 82: PUT /Subscriptions/{id}/Terminate Parameters

5.3.7.9 DELETE /Subscriptions/{id}/Delete

Deletes the *Subscription* with the *Id* provided. All *Notification* and *NotificationItem* created from the *Subscription* will also be deleted.

Parameter	Description	Parameter Type	Data Type
-----------	-------------	----------------	-----------

id	<i>Id</i> of the <i>Subscription</i> to be deleted. Must be double URL encoded.	<i>Path</i>	<i>String</i>
----	---	-------------	---------------

Table 83: DELETE /Subscriptions/{id}/Delete Parameters

5.3.7.10 DELETE /Subscriptions/{id}/DeleteNotifications

Deletes [Notification](#) for the *Subscription* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Subscription</i> which <i>Notification</i> are to be deleted. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 84: DELETE /Subscriptions/{id}/DeleteNotifications Parameters

5.3.8 SystemEvents

A *SystemEvent* represents the occurrence of a server specific event.

The properties of a *SystemEvent* are listed in Table 85.

Property	Description	Data Type
Id	Case sensitive identifier for the <i>AlarmEvent</i> .	<i>String</i>
Message	Human readable message for the <i>AlarmEvent</i> .	<i>String</i>
OccuredOn	Date <i>Time</i> in UTC when the <i>Alarm</i> most recently transitioned into and "alarmed" state - that is not in the "Normal" state.	<i>DateTime</i>
Priority	Priority value of the <i>AlarmEvent</i> .	<i>Integer</i>
SourceId	Id for a server specific "source" of the <i>SystemEvent</i> .	<i>String</i>
SourceName	<i>Name</i> for the server specific "source" of the <i>SystemEvent</i> .	<i>String</i>

Table 85: SystemEvent Model

5.3.8.1 GET /SystemEvents

Returns all *SystemEvent* which fit the filter criteria supplied. All filter parameters are optional.

SystemEvent will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
priorityFrom	Return only <i>SystemEvent</i> with a <i>Priority</i> greater or equal to the value supplied.	<i>Query</i>	<i>Integer</i>
priorityTo	Return only <i>SystemEvent</i> with a <i>Priority</i> less or equal to the value	<i>Query</i>	<i>Integer</i>

	supplied.		
state	Return only <i>SystemEvent</i> with this <i>State</i> value.	Query	Enumeration (See Table 1)
acknowledgeable	Return only <i>SystemEvent</i> with this <i>Acknowledgeable</i> value.	Query	Enumeration (See Table 1)
type	Return only <i>SystemEvent</i> with this <i>Type</i> value.	Query	String
occurredOnOrAfter	Return only <i>SystemEvent</i> with <i>OccurredOn</i> which is on or after this value.	Query	DateTime
occurredBefore	Return only <i>SystemEvent</i> with <i>OccurredOn</i> which is before this value.	Query	DateTime
orderBy	<i>SystemEvent</i> will be returned in this order. <i>OccurredOnAscending</i> by default.	Query	Enumeration OccurredOnAscending =0 OccurredOnDescending =1
take	Number of <i>SystemEvent</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>SystemEvent</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 86: GET /SystemEvents Parameters

MongooseEwsRestProvider does not support this action.

5.3.8.2 GET /SystemEvents/{id}

Returns the *SystemEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>SystemEvent</i> requested. Must be double URL encoded.	Path	String

Table 87: GET /SystemEvents/{id} Parameters

MongooseEwsRestProvider does not support this action.

5.3.8.3 GET /SystemEvents/{id}/Message

Returns the *Message* property for the *SystemEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	<i>Id</i> of the <i>SystemEvent</i> whose <i>Message</i> property is to be returned. Must be	Path	String

	double URL encoded.		
--	---------------------	--	--

Table 88: GET /SystemEvents/{id}/Message Parameters

MongooseEwsRestProvider does not support this action.

5.3.8.4 GET /SystemEvents/{id}/OccurredOn

Returns the *OccurredOn* property for the *SystemEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>SystemEvent</i> whose <i>OccurredOn</i> property is to be returned. Must be double URL encoded.	Path	String

Table 89: GET /SystemEvents/{id}/OccurredOn Parameters

MongooseEwsRestProvider does not support this action.

5.3.8.5 GET /SystemEvents/{id}/Priority

Returns the *Priority* property for the *SystemEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>SystemEvent</i> whose <i>Priority</i> property is to be returned. Must be double URL encoded.	Path	String

Table 90: GET /SystemEvents/{id}/Priority Parameters

MongooseEwsRestProvider does not support this action.

5.3.8.6 GET /SystemEvents/{id}/SourceId

Returns the *SourceId* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>SystemEvent</i> whose <i>SourceId</i> property is to be returned. Must be double URL encoded.	Path	String

Table 91: GET /SystemEvents/{id}/Priority Parameters

MongooseEwsRestProvider does not support this action.

5.3.8.7 GET /SystemEvents/{id}/SourceName

Returns the *SourceName* property for the *SystemEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>SystemEvent</i> whose <i>SourceName</i> property is to be returned. Must be double URL encoded.	Path	String

Table 92: GET /SystemEvents/{id}/Priority Parameters

MongooseEwsRestProvider does not support this action.

5.3.8.8 GET /SystemEvents/Type

Returns the *Type* property for the *SystemEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>SystemEvent</i> whose <i>Type</i> property is to be returned. Must be double URL encoded.	Path	String

Table 93: GET /SystemEvents/{id}/Type Parameters

MongooseEwsRestProvider does not support this action.

5.3.9 SystemEventTypes

A *SystemEventType* represents a server specific a Type for a [SystemEvent](#).

The properties of a *SystemEventType* are listed in Table 94.

Property	Description	Data Type
Description	Optional description of the <i>AlarmEventType</i> .	String
Id	Case sensitive identifier for the <i>AlarmEventType</i> .	String
Name	Name of the <i>AlarmEventType</i> .	String

Table 94: SystemEventType Model

5.3.9.1 GET /SystemEventTypes

Returns all *SystemEventType* which fit the filter criteria supplied. All filter parameters are optional. *SystemEventType* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
name	Return only <i>SystemEventType</i> whose <i>Name</i> contains this value.	Query	String
orderBy	<i>SystemEventType</i> will be returned in this order. <i>NameAscending</i> by default.	Query	Enumeration NameAscending=0 NameDescending=1

take	Number of <i>SystemEventType</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>SystemEventType</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 95: GET /SystemEventTypes Parameters

MongooseEwsRestProvider does not support this action.

5.3.9.2 GET / SystemEventTypes/{id}

Returns the *SystemEventType* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>SystemEventType</i> requested. Must be double URL encoded.	Path	String

Table 96: GET /SystemEventTypes/{id} Parameters

MongooseEwsRestProvider does not support this action.

5.3.9.3 GET / SystemEventTypes/{id}/Description

Returns the *Description* property for the *SystemEventType* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>SystemEventType</i> whose <i>Description</i> property is to be returned. Must be double URL encoded.	Path	String

Table 97: GET /SystemEventTypes/{id}/Description Parameters

MongooseEwsRestProvider does not support this action.

5.3.9.4 GET / SystemEventTypes/{id}/Name

Returns the *Name* property for the *SystemEventType* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>SystemEventType</i> whose <i>Name</i> property is to be returned. Must be double URL encoded.	Path	String

Table 98: GET /SystemEventTypes/{id}/Name Parameters

MongooseEwsRestProvider does not support this action.

5.3.10 Trends

A *Trend* references a specific [Value](#) for which historical value "trending" has been enabled. The existence of a *Trend* does not imply that history exists; only that history will be captured for a *Value*.

Property	Description	Data Type
Description	Optional description of the <i>Trend</i> .	<i>String</i>
Id	Case sensitive identifier for the <i>Trend</i> .	<i>String</i>
Name	Name of the <i>Trend</i> .	<i>String</i>
ParentId	Id of the <i>Container</i> which the <i>Trend</i> is found in.	<i>String</i>
Type	Type of the associated <i>Value</i> .	<i>Enumeration</i> DateTime=0 Boolean=1 String=2 Double=3 Long=4 Integer=5 Duration=6 See Section 4.3.2 from the EWS Specification for more details.
Unit	Unit for the associated <i>Value</i> .	<i>String</i>
ValueId	Id of the <i>Value</i> for which trending data will be captured.	<i>String</i>

Table 99: *Trend* Model

5.3.10.1 *POST /Trends/Create*

Creates a new *Trend* and returns it.

Parameter	Description	Parameter Type	Data Type
newItem	Definition of how the <i>Trend</i> will be created.	<i>Body</i>	<i>Object</i>
Description	<i>Description</i> for the new <i>Trend</i> .		<i>String</i>
Id	<i>Id</i> for the new <i>Trend</i> .		<i>String</i>
Name	<i>Name</i> for the new <i>Trend</i> .		<i>String</i>
ParentId	Optional <i>Id</i> of the <i>Container</i> the new <i>Trend</i> will be put into.		<i>String</i>
Type	<i>Type</i> for the new <i>Trend</i> .		<i>Enumeration</i> See Table 99
ValueId	<i>Id</i> for the <i>Value</i> for which historical trends will be captured.		<i>String</i>

Table 100: *POST /Containers/Create* Parameters

SoapEwsRestProvider does not support this action.

5.3.10.2 GET /Trends

Returns all *Trend* which fit the filter criteria supplied. All filter parameters are optional. *Trend* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
name	Return only <i>Trend</i> whose name contains this value.	Query	String
orderBy	<i>Trend</i> will be returned in this order. <i>NameAscending</i> by default.	Query	Enumeration NameAscending=0 NameDescending=1
take	Number of <i>Trend</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>Trend</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 101: GET /Trends Parameters

SoapEwsRestProvider does not support this action.

5.3.10.3 GET /Trends/{id}

Returns the *Trend* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> requested. Must be double URL encoded.	Path	String

Table 102: GET /Trends/{id} Parameters

5.3.10.4 GET /Trends/{id}/Description

Returns the *Description* property for the *Trend* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Description</i> property is to be returned. Must be double URL encoded.	Path	String

Table 103: GET /Trends/{id}/Description Parameters

5.3.10.5 PUT /Trends/{id}/Description

Modifies the *Description* property for the *Trend* with the *Id* provided to the new value supplied. Returns the modified *Trend*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Description</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Description</i> value.	<i>Body</i>	<i>String</i>

Table 104: PUT /Trends/{id}/Description Parameters

SoapEwsRestProvider does not support this action.

5.3.10.6 GET /Trends/{id}/Name

Returns the *Name* property for the *Trend* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Name</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 105: GET /Trends/{id}/Name Parameters

5.3.10.7 PUT /Trends/{id}/Name

Modifies the *Name* property for the *Trend* with the *Id* provided to the new value supplied. Returns the modified *Trend*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Name</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Name</i> value.	<i>Body</i>	<i>String</i>

Table 106: PUT /Trends/{id}/Name Parameters

SoapEwsRestProvider does not support this action.

5.3.10.8 GET /Trends/{id}/Parent

Returns the parent [Container](#) for the *Trend* with the *Id* supplied.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose parent <i>Container</i> is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 107: GET /Trends/{id}/Parent Parameters

5.3.10.9 GET /Trends/{id}/ParentId

Returns the *ParentId* property for the *Trend* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>ParentId</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 108: GET /Trends/{id}/ParentId Parameters

5.3.10.10 PUT /Containers/{id}/ParentId

Modifies the *ParentId* property for the *Trend* with the *Id* provided to the new value supplied. Returns the modified *Trend*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>ParentId</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>ParentId</i> value.	<i>Body</i>	<i>String</i>

Table 109: PUT /Trends/{id}/ParentId Parameters

SoapEwsRestProvider does not support this action.

5.3.10.11 GET /Trends/{id}/Type

Returns the *Type* property for the *Trend* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Type</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 110: GET /Trends/{id}/Type Parameters

5.3.10.12 GET /Trends/{id}/Unit

Returns the *Unit* property for the *Trend* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Unit</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 111: GET /Trends/{id}/Unit Parameters

5.3.10.13 GET /Trends/{id}/Value

Returns the [Value](#) the *Trend* with the *Id* supplied.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Value</i> is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 112: GET /Trends/{id}/Value Parameters

5.3.10.14 GET /Trends/{id}/ValueId

Returns the *ValueId* property for the *Trend* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>ValueId</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 113: GET /Trends/{id}/ValueId Parameters

5.3.10.15 PATCH /Trends/{id}/Update

Modifies the *Trend* with the *Id* provided to the values supplied. Returns the modified *Trend*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
updatedItem	Values to update the <i>Trend</i> with.	<i>Body</i>	<i>Object</i>
Description	New <i>Description</i> value.		<i>String</i>
Id	New <i>Id</i> value.		<i>String</i>
Name	New <i>Name</i> value.		<i>String</i>
ParentId	New <i>ParentId</i> value.		<i>String</i>

Table 114: PATCH /Trends/{id}/Update Parameters

SoapEwsRestProvider does not support this action.

5.3.10.16 PUT /Trends/{id}/Id

Modifies the *Id* property for the *Trend* with the *Id* provided to the new value supplied. Returns the modified *Trend*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> whose <i>Id</i> property is	<i>Path</i>	<i>String</i>

	to be modified. Must be double URL encoded.		
newValue	New <i>Id</i> value.	Body	String

Table 115: PUT /Trends/{id}/Id Parameters

SoapEwsRestProvider does not support this action.

5.3.10.17 DELETE /Trends/{id}/Delete

Deletes the *Trend* with the *Id* provided. All [TrendSample](#) associated with the *Trend* will also be deleted.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> to be deleted. Must be double URL encoded.	Path	String

Table 116: DELETE /Containers/{id}/Delete Parameters

SoapEwsRestProvider does not support this action.

5.3.10.18 DELETE /Trends/{id}/DeleteTrendSamples

Deletes a range of [TrendSample](#) associated with the *Trend* with the *Id* provided. Open ended ranges are supported. If *sampleDateRange* is not included all *TrendSample* will be deleted for the *Trend*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Trend</i> for which <i>TrendSample</i> are to be deleted. Must be double URL encoded.	Path	String
sampleDateRange	Only <i>TrendSample</i> with a <i>SampleDate</i> in this range will be deleted.	Body	Object
Before	DateTime in UTC		DateTime
OnOrAfter	DateTime in UTC		DateTime

Table 117: DELETE /Trends/{id}/DeleteTrendSamples Parameters

SoapEwsRestProvider does not support this action.

5.3.11 TrendSamples

A *TrendSample* represents the value and state for [Value](#) at a given point in time belonging to a specific [Trend](#).

Property	Description	Data Type
Id	Case sensitive identifier for the <i>Value</i> .	String
State	State of the <i>Trend's Value</i> at the time of the	Enumeration

	sample.	Good=0 Uncertain=1 Forced=2 Offline=3 Error=4 See section 4.3.4 of the EWS Specification for more details.
SampleDate	DateTime in UTC when the sample was created.	<i>DateTime</i>
TrendId	Id of the <i>Trend</i> which this is a sample for.	<i>String</i>
Value	Value of the <i>Trend's Value</i> at the time of the sample.	<i>String</i>

 Table 118: *TrendSample* Model

5.3.11.1 GET /TrendSamples

Returns all *TrendSample* which fit the filter criteria supplied. All filter parameters are optional except *trendId*. *TrendSample* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
alarmId	Return only <i>AlarmEvent</i> for the referenced Alarm .	<i>Query</i>	<i>String</i>
trendId	The <i>Id</i> of the <i>Trend</i> which <i>TrendSample</i> are to be returned for. This parameter is not optional.	<i>Query</i>	<i>String</i>
state	Return only <i>TrendSample</i> with this <i>State</i> value.	<i>Query</i>	<i>Enumeration</i> (See Table 1)
sampleOnOrAfter	Return only <i>TrendSample</i> with <i>SampleDate</i> which is on or after this value.	<i>Query</i>	<i>DateTime</i>
sampledBefore	Return only <i>TrendSample</i> with <i>SampleDate</i> which is before this value.	<i>Query</i>	<i>DateTime</i>
orderBy	<i>TrendSample</i> will be returned in this order. <i>SampleDateAscending</i> by default.	<i>Query</i>	<i>Enumeration</i> SampleDateAscending=0 SampleDateDescending=1
take	Number of <i>TrendSample</i> that should be returned. 200 by default.	<i>Query</i>	<i>Integer</i>
skip	Number of <i>TrendSample</i> that should be skipped before items are returned. 0 by default.	<i>Query</i>	<i>Integer</i>

Table 119: GET /TrendSamples Parameters

5.3.11.2 GET /TrendSamples/{id}

Returns the *TrendSample* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>TrendSample</i> requested. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 120: GET /TrendSamples/{id} Parameters

5.3.11.3 GET /TrendSamples/{id}/State

Returns the *State* property for the *TrendSample* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>TrendSample</i> whose <i>State</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 121: GET /AlarmEvents/{id}/State Parameters

5.3.11.4 GET /TrendSamples/{id}/SampleDate

Returns the *SampleDate* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>TrendSample</i> whose <i>SampleDate</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 122: GET /TrendSamples/{id}/SampleDate Parameters

5.3.11.5 GET /TrendSamples/{id}/Trend

Returns the *Trend* for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>TrendSample</i> whose <i>Trend</i> is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 123: GET /TrendSamples/{id}/Trend Parameters

5.3.11.6 GET /TrendSamples/{id}/TrendId

Returns the *TrendId* property for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
-----------	-------------	----------------	-----------

Id	Id of the <i>TrendSample</i> whose <i>TrendId</i> property is to be returned. Must be double URL encoded.	Path	String
----	---	------	--------

Table 124: GET /TrendSamples/{id}/TrendId Parameters

5.3.11.7 GET /TrendSamples/{id}/Value

Returns the [Value](#) for the *AlarmEvent* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
Id	Id of the <i>TrendSample</i> whose <i>Value</i> is to be returned. Must be double URL encoded.	Path	String

Table 125: GET /TrendSamples/{id}/Value Parameters

5.3.12 Values

A *Value* represents a singular point value. Point values can be physical values from a sensor or controller, internal properties such as a software version, or any other value an implementing server decides to expose.

Property	Description	Data Type
Description	Optional description of the <i>Value</i> .	String
Forceable	Force restrictions for the <i>Value</i> .	Enumeration NotForceable=0 Forceable=1 See section 4.3.5 of the EWS Specification for more details.
Id	Case sensitive identifier for the <i>Value</i> .	String
Name	Name of the <i>Value</i> .	String
ParentId	Id of the Container which the <i>Value</i> is found in.	String
State	Validity of the <i>Value</i> .	Enumeration Good=0 Uncertain=1 Forced=2 Offline=3 Error=4 See section 4.3.4 of the EWS Specification for more details.
Type	Implied type of the <i>Value</i> value property.	Enumeration DateTime=0 Boolean=1 String=2 Double=3 Long=4

		Integer=5 Duration=6 See Section 4.3.2 from the EWS Specification for more details.
Unit	Unit of the <i>Value</i> .	<i>String</i>
Value	Value of the <i>Value</i> .	<i>String</i>
Writeable	Set restrictions for the <i>Value</i> .	<i>Enumeration</i> ReadOnly=0 Writeable=1 See section 4.3.3 of the EWS Specification for more details.

 Table 126: *Value* Model

5.3.12.1 *POST /Values/Create*

Creates a new *Value* and returns it.

Parameter	Description	Parameter Type	Data Type
newItem	Definition of how the <i>Value</i> will be created.	<i>Body</i>	<i>Object</i>
Description	<i>Description</i> for the new <i>Value</i> .		<i>String</i>
Forceable	<i>Forceable</i> for the new <i>Value</i> .		<i>Enumeration</i>
Id	<i>Id</i> for the new <i>Value</i> .		<i>String</i>
Name	<i>Name</i> for the new <i>Value</i> .		<i>String</i>
ParentId	Optional <i>Id</i> of the <i>Value</i> the new <i>Value</i> will be put into.		<i>String</i>
State	<i>State</i> for the new <i>Value</i> .		<i>Enumeration</i>
Type	<i>Type</i> for the new <i>Value</i> .		<i>Enumeration</i> See Table 47
Unit	<i>Unit</i> for the new <i>Value</i>		<i>String</i>
Value	<i>Value</i> for the new <i>Value</i>		<i>String</i>
Writeable	<i>Writeable</i> for the new <i>Value</i> .		<i>Enumeration</i>

 Table 127: *POST /Values/Create* Parameters

SoapEwsRestProvider does not support this action.

5.3.12.2 *GET /Values*

Returns all *Value* which fit the filter criteria supplied. All filter parameters are optional. *Value* will be ordered and paged as requested.

Parameter	Description	Parameter Type	Data Type
name	Return only <i>Value</i> whose name	<i>Query</i>	<i>String</i>

	contains this value.		
type	Return only <i>Value</i> with this <i>Type</i> value.	Query	Enumeration See Table 126
orderBy	<i>Value</i> will be returned in this order. <i>NameAscending</i> by default.	Query	Enumeration NameAscending=0 NameDescending=1
take	Number of <i>Value</i> that should be returned. 200 by default.	Query	Integer
skip	Number of <i>Value</i> that should be skipped before items are returned. 0 by default.	Query	Integer

Table 128: GET /Values Parameters

SoapEwsRestProvider does not support this action.

5.3.12.3 GET /Values/{id}

Returns the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> requested. Must be double URL encoded.	Path	String

Table 129: GET /Values/{id} Parameters

5.3.12.4 GET /Values/{id}/Description

Returns the *Description* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Description</i> property is to be returned. Must be double URL encoded.	Path	String

Table 130: GET /Values/{id}/Description Parameters

5.3.12.5 PUT /Values/{id}/Description

Modifies the *Description* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Description</i> property is to be modified. Must be double URL encoded.	Path	String

newValue	New <i>Description</i> value.	Body	String
----------	-------------------------------	------	--------

Table 131: PUT /Values/{id}/Description Parameters

SoapEwsRestProvider does not support this action.

5.3.12.6 GET /Values/{id}/Forceable

Returns the *Forceable* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Forceable</i> property is to be returned. Must be double URL encoded.	Path	String

Table 132: GET /Values/{id}/Forceable Parameters

5.3.12.7 PUT /Values/{id}/Forceable

Modifies the *Forceable* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Forceable</i> property is to be modified. Must be double URL encoded.	Path	String
newValue	New <i>Forceable</i> value.	Body	String

Table 133: PUT /Values/{id}/Forceable Parameters

SoapEwsRestProvider does not support this action.

5.3.12.8 GET /Values/{id}/Name

Returns the *Name* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Name</i> property is to be returned. Must be double URL encoded.	Path	String

Table 134: GET /Values/{id}/Name Parameters

5.3.12.9 PUT /Containers/{id}/Name

Modifies the *Name* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
-----------	-------------	----------------	-----------

id	<i>Id of the Value whose Name property is to be modified. Must be double URL encoded.</i>	<i>Path</i>	<i>String</i>
newValue	<i>New Name value.</i>	<i>Body</i>	<i>String</i>

Table 135: PUT /Values/{id}/Name Parameters

SoapEwsRestProvider does not support this action.

5.3.12.10 GET /Values/{id}/Parent

Returns the parent [Container](#) for the *Value* with the *Id* supplied.

Parameter	Description	Parameter Type	Data Type
id	<i>Id of the Value whose parent Container is to be returned. Must be double URL encoded.</i>	<i>Path</i>	<i>String</i>

Table 136: GET /Values/{id}/Parent Parameters

5.3.12.11 GET /Values/{id}/ParentId

Returns the *ParentId* property for the *Container* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id of the Value whose ParentId property is to be returned. Must be double URL encoded.</i>	<i>Path</i>	<i>String</i>

Table 137: GET /Values/{id}/ParentId Parameters

5.3.12.12 PUT /Values/{id}/ParentId

Modifies the *ParentId* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id of the Value whose ParentId property is to be returned. Must be double URL encoded.</i>	<i>Path</i>	<i>String</i>

Table 138: PUT /Values/{id}/ParentId Parameters

SoapEwsRestProvider does not support this action.

5.3.12.13 GET /Values/{id}/State

Returns the *State* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	Id of the <i>State</i> whose <i>Type</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 139: GET /Values/{id}/State Parameters

5.3.12.14 PUT /Values/{id}/State

Modifies the *State* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>State</i> whose <i>Type</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>State</i> value.	<i>Body</i>	<i>Enumeration</i> See Table 47

Table 140: PUT /Values/{id}/State Parameters

SoapEwsRestProvider does not support this action.

5.3.12.15 GET /Values/{id}/Type

Returns the *Type* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Type</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 141: GET /Values/{id}/Type Parameters

5.3.12.16 PUT /Values/{id}/Type

Modifies the *Type* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Type</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Type</i> value.	<i>Body</i>	<i>Enumeration</i> See Table 47

Table 142: PUT /Values/{id}/Type Parameters

SoapEwsRestProvider does not support this action.

5.3.12.17 GET /Values/{id}/Unit

Returns the *Unit* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Unit</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 143: GET /Values/{id}/Unit Parameters

5.3.12.18 PUT /Values/{id}/Unit

Modifies the *Unit* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Unit</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Unit</i> value.	<i>Body</i>	<i>Enumeration</i> See Table 47

Table 144: PUT /Values/{id}/Unit Parameters

SoapEwsRestProvider does not support this action.

5.3.12.19 GET /Values/{id}/Value

Returns the *Value* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Value</i> property is to be returned. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 145: GET /Values/{id}/Value Parameters

5.3.12.20 PUT /Values/{id}/Value

Modifies the *Value* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Value</i> property is to be modified. Must be	<i>Path</i>	<i>String</i>

	double URL encoded.		
newValue	New <i>Value</i> value.	Body	Enumeration See Table 47

Table 146: PUT /Values/{id}/Value Parameters

5.3.12.21 GET /Values/{id}/Writeable

Returns the *Writeable* property for the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Writeable</i> property is to be returned. Must be double URL encoded.	Path	String

Table 147: GET /Values/{id}/Writeable Parameters

5.3.12.22 PUT /Values/{id}/Writeable

Modifies the *Writeable* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Writeable</i> property is to be modified. Must be double URL encoded.	Path	String
newValue	New <i>Writeable</i> value.	Body	String

Table 148: PUT /Values/{id}/Writeable Parameters

SoapEwsRestProvider does not support this action.

5.3.12.23 PATCH /Values/{id}/Update

Modifies the *Value* with the *Id* provided to the values supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> to be modified. Must be double URL encoded.	Path	String
updatedItem	Values to update the <i>Value</i> with.	Body	Object
Description	New <i>Description</i> value.		String
Forceable	New <i>Forceable</i> value.		Enumeration See Table 126
Id	New <i>Id</i> value.		String
Name	New <i>Name</i> value.		String

ParentId	New <i>ParentId</i> value.		<i>String</i>
Type	New <i>Type</i> value.		<i>Enumeration</i> See Table 126
Unit	New <i>Unit</i> value.		<i>String</i>
Value	New <i>Value</i> value.		<i>String</i>
Writeable	New <i>Writeable</i> value.		<i>Enumeration</i> See Table 126

Table 149: PATCH /Values/{id}/Update Parameters

SoapEwsRestProvider does not support this action.

5.3.12.24 PUT /Values/{id}/Id

Modifies the *Id* property for the *Value* with the *Id* provided to the new value supplied. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> whose <i>Id</i> property is to be modified. Must be double URL encoded.	<i>Path</i>	<i>String</i>
newValue	New <i>Id</i> value.	<i>Body</i>	<i>String</i>

Table 150: PUT /Values/{id}/Id Parameters

SoapEwsRestProvider does not support this action.

5.3.12.25 PUT /Values/{id}/Force

Forces the *Value* with the *Id* provided. After this action the *Value* will have a *State of Forced* and a *Value* equal to the *forcedValue* parameter. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> to be forced. Must be double URL encoded.	<i>Path</i>	<i>String</i>
forcedValue	<i>Value</i> which the <i>Value</i> will be forced to.	<i>Body</i>	<i>String</i>

Table 151: PUT /Values/{id}/Force Parameters

5.3.12.26 PUT /Values/{id}/Unforce

Unforces the *Value* with the *Id* provided. After this action the *Value* will have a *State* and *Value* which depends on configured Data Provider. Returns the modified *Value*.

Parameter	Description	Parameter Type	Data Type
-----------	-------------	----------------	-----------

id	Id of the <i>Value</i> to be unforced. Must be double URL encoded.	<i>Path</i>	<i>String</i>
----	--	-------------	---------------

Table 152: PUT /Values/{id}/Unforce Parameters

5.3.12.27 DELETE /Values/{id}/Delete

Deletes the *Value* with the *Id* provided.

Parameter	Description	Parameter Type	Data Type
id	<i>Id</i> of the <i>Value</i> to be deleted. Must be double URL encoded.	<i>Path</i>	<i>String</i>

Table 153: DELETE /Values/{id}/Delete Parameters

SoapEwsRestProvider does not support this action.

6 Swagger

The architecture for the SmartConnector REST Extensions integrates [Swagger](#) to provide a rich self-documenting API. What is Swagger? From the Swagger website itself:

“The goal of Swagger™ is to define a standard, language-agnostic interface to REST APIs which allows both humans and computers to discover and understand the capabilities of the service without access to source code, documentation, or through network traffic inspection. When properly defined via Swagger, a consumer can understand and interact with the remote service with a minimal amount of implementation logic. Similar to what interfaces have done for lower-level programming, Swagger removes the guesswork in calling the service.

Technically speaking - Swagger is a formal specification surrounded by a large ecosystem of tools, which includes everything from front-end user interfaces, low-level code libraries and commercial API management solutions.”

Accessing the Swagger page for any SmartConnector REST Endpoint is quite easy. From the SmartConnector Portal Status page, view all Active Endpoints as shown in Figure 4. If you do not see a link then edit the configuration as [described above](#) and ensure that “Server Swagger Metadata” is set to “True”.

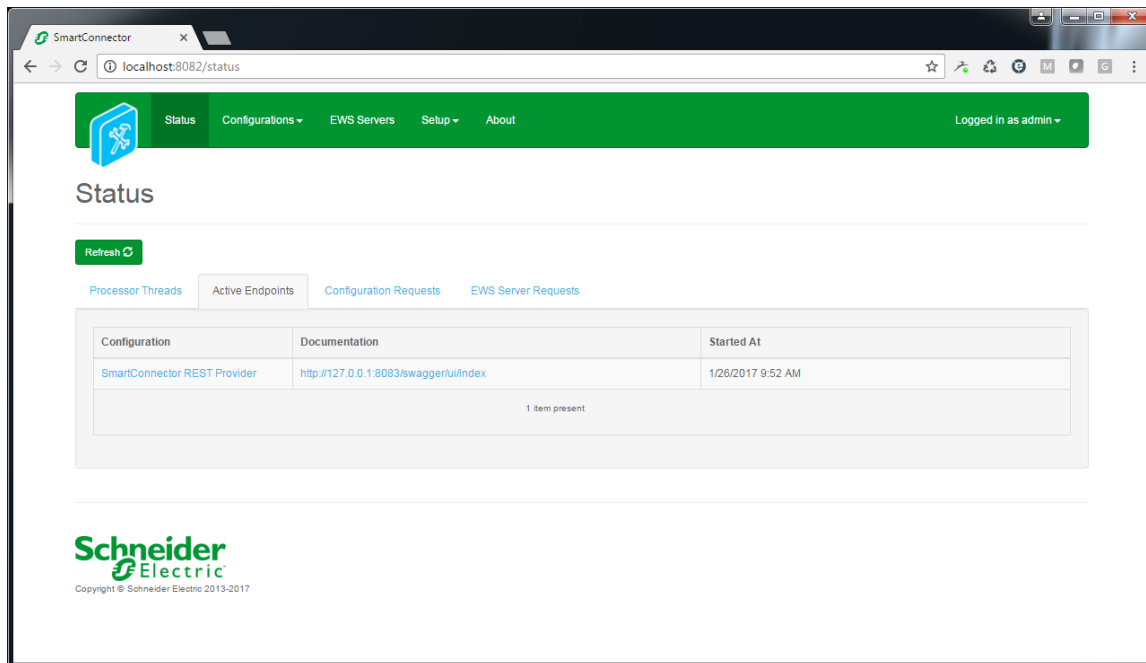


Figure 4: Portal - Status Page

Clicking the “Documentation” link for any provisioned REST Endpoint will open the Swagger page in a new browser tab (Figure 5).

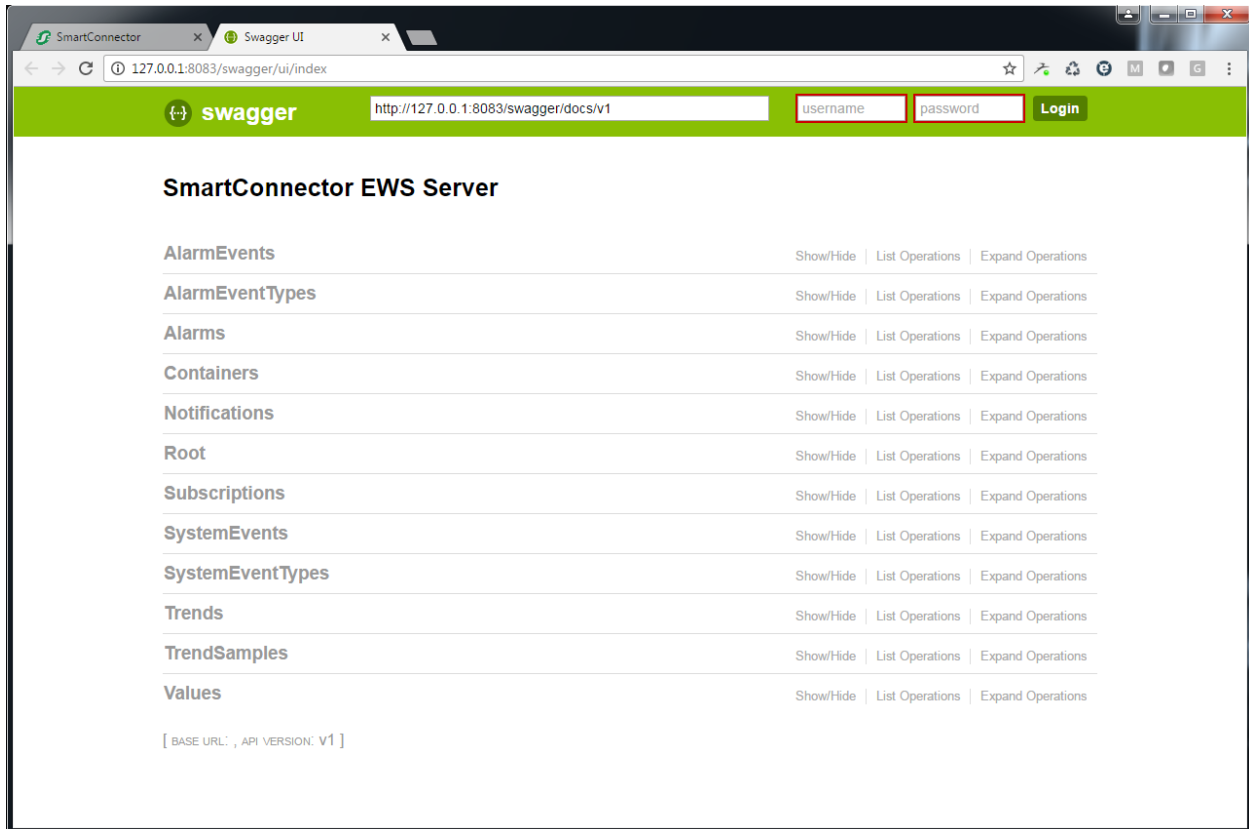


Figure 5: Swagger Page

6.1 Using Swagger

While using Swagger to exercise the RESTful EWS Gateway is intuitive, there are some customizations and tips to know to make your experience as easy as possible. These are discussed in the following sections.

6.1.1 Authentication

Before any commands can be attempted, an authentication Bearer Token is required. To facilitate authentication, the generic Swagger page has been customized for SmartConnector. In the upper right corner of the page you will find “username” and “password” input fields and a “Login” button. The red border around the input fields, as shown in Figure 6, indicates that authentication has not occurred and no Bearer Token is available.

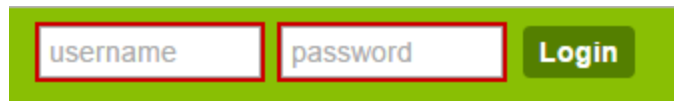


Figure 6: Swagger – Not Authenticated

The credentials required depend on which [Data Provider](#) is being used. For example, if you are using the *MongooseEwsRestProvider* the credentials required are those of any User account in the referenced

Native EWS Server. Alternatively, if you are using the *SoapEwsRestProvider* the credentials required are those of the “EWS User” for the referenced server.

When a valid username and password are entered, clicking the “Login” button will request a valid Bearer Token from the REST Endpoint. Successful authentication will be indicated with green borders around the input fields as shown in Figure 7. If authentication fails, the Swagger page will inform you with a dialog warning. Verify your credentials and try again.



Figure 7: Swagger - Authenticated

After authentication, Swagger will automatically add the current Bearer Token, in an Authentication Header, for all actions shown on the Swagger page. This will continue for the duration of the Bearer Token’s life (or a full page refresh is performed). If the Bear Token expires, you simply need to re-authenticate by clicking the Login button again.

6.1.2 Issuing Requests

As you navigate the Swagger page, you will become familiar with the pertinent aspects of the [resources and actions](#) provided in the RESTful EWS Gateway.

For example, Figure 8 shows the [GET /Root](#) resource action for the RESTful EWS Gateway. This action requests the “root” *Container* resource.

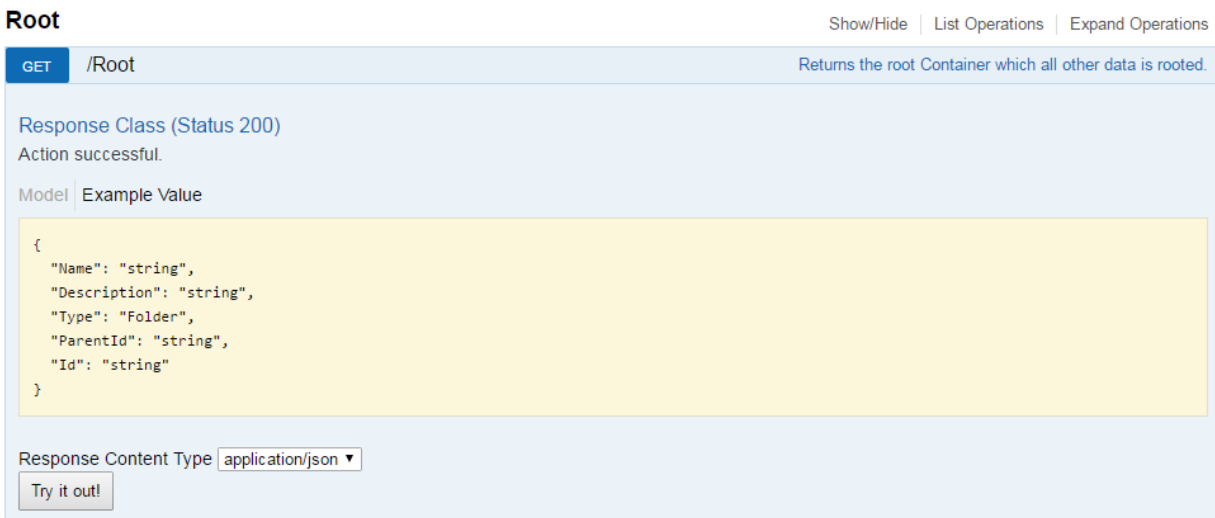
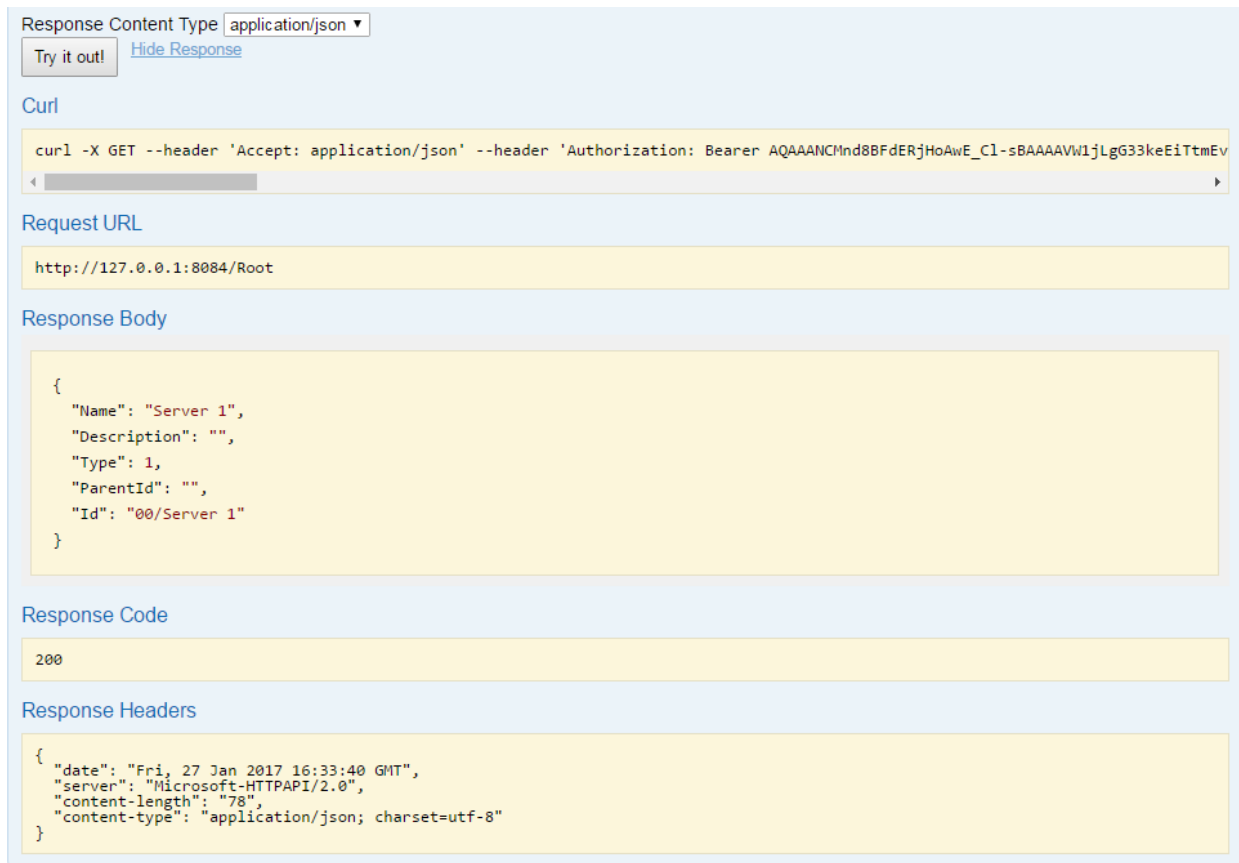


Figure 8: Swagger – GET /Root

The top portion displays the expected success status code for the action (Status 200) and “Example Value” and “Model” synopsis of the data to be returned. Clicking the “Try it out!” button will issue the command and display the response as seen in Figure 9.



The screenshot displays the Swagger UI interface for a RESTful API. At the top, the 'Response Content Type' is set to 'application/json'. Below this, there are buttons for 'Try it out!' and 'Hide Response'. The 'Curl' section shows the command: `curl -X GET --header 'Accept: application/json' --header 'Authorization: Bearer AQAAANCnd8BFdERjHoAwE_C1-sBAAAAVW1jLgG33keEiTtmEv'`. The 'Request URL' is `http://127.0.0.1:8084/Root`. The 'Response Body' contains a JSON object:

```
{
  "Name": "Server 1",
  "Description": "",
  "Type": 1,
  "ParentId": "",
  "Id": "00/Server 1"
}
```

. The 'Response Code' is 200. The 'Response Headers' section shows:

```
{
  "date": "Fri, 27 Jan 2017 16:33:40 GMT",
  "server": "Microsoft-HTTPAPI/2.0",
  "content-length": "78",
  "content-type": "application/json; charset=utf-8"
}
```

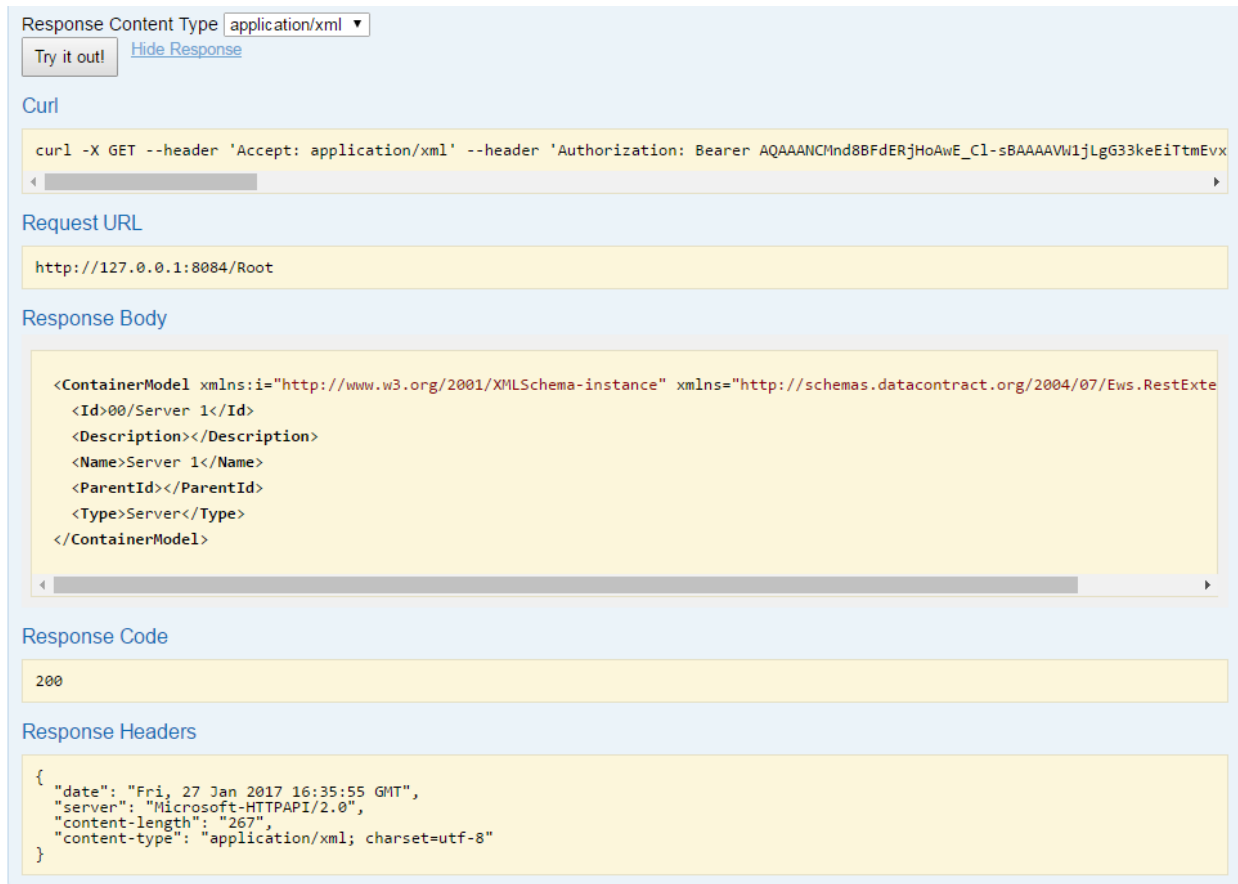
Figure 9: Swagger – GET /Root Response (JSON)

Of note in the response section are the [Curl](#) command, Request URL, Response Body and Response Code, and Response Headers.

- **Curl** – Shows how the request was made, including all header and parameter information. Examining the link you will see the “Accept” and “Authorization” header values along with the URL Swagger requested.
- **Request URL** – Shows the URL aspect only and is helpful when parameters are added to the request.
- **Response Body** – The actual body of the response. In our case, we see the root [Container](#) in JSON format (because we issued the request with an “Accept” header of “application/json”).
- **Response Code** – The HTTP code of the response.
- **Response Headers** – Shows all response headers from the server. This will where any custom [response headers](#) will be included when appropriate.

If we change the content type to “application/xml” and click “Try it out!” again, the server will respond in much the same manner as shown in Figure 10. Note that the Response Body and Response Header for “content-type” now show XML format. Whether JSON or XML format is used is largely the

preference of the consumer and has no functional difference other than XML payload will be larger due to the additional markup included.



The screenshot shows the Swagger UI interface for a RESTful API. The 'Response Content Type' is set to 'application/xml'. The 'Request URL' is 'http://127.0.0.1:8084/Root'. The 'Response Code' is '200'. The 'Response Headers' are:

```
{
  "date": "Fri, 27 Jan 2017 16:35:55 GMT",
  "server": "Microsoft-HTTPAPI/2.0",
  "content-length": "267",
  "content-type": "application/xml; charset=utf-8"
}
```

The 'Response Body' is an XML document:

```
<ContainerModel xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://schemas.datacontract.org/2004/07/Ews.RestExt">
  <Id>00/Server 1</Id>
  <Description></Description>
  <Name>Server 1</Name>
  <ParentId></ParentId>
  <Type>Server</Type>
</ContainerModel>
```

Figure 10: Swagger – GET /Root Response (XML)

6.1.3 Parameters

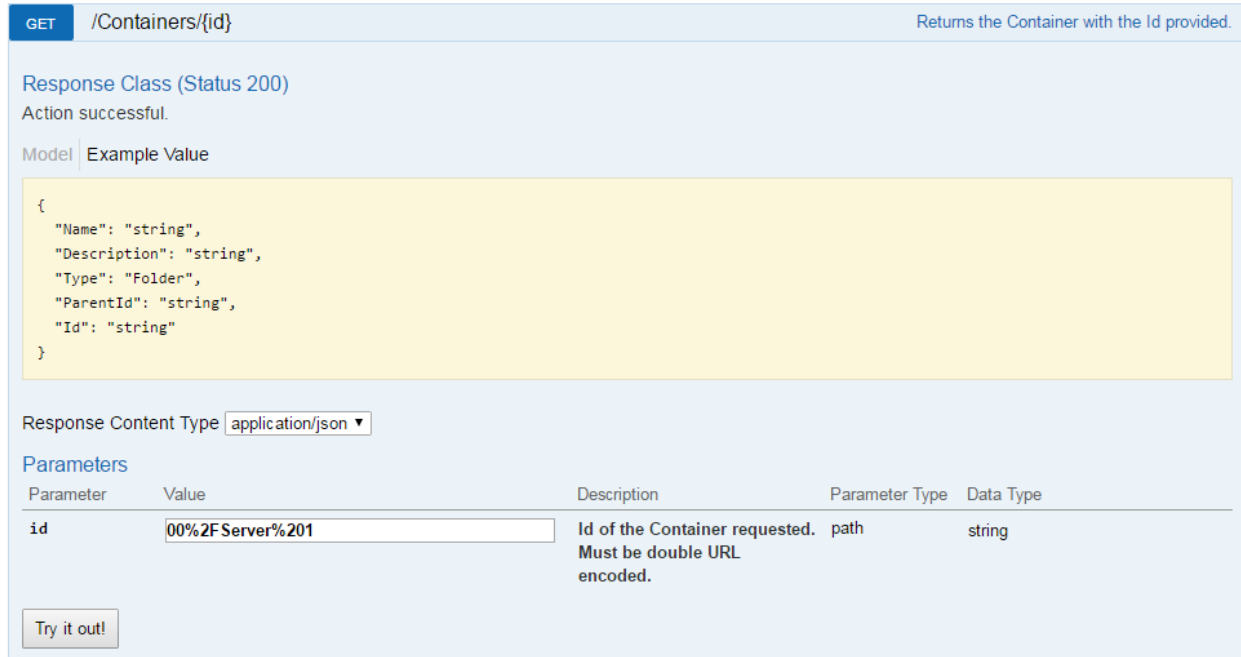
Reviewing the [resources and actions](#) supported by the RESTful EWS Gateway, you will notice that there are three types of parameters which actions can use: Path, Query, and Body.

6.1.3.1 Path Parameters

Path parameters are those that are part of the route path itself. Not all actions in the RESTful EWS Gateway make use of path parameters. Those that do, only use a single path parameter, the *Id* of the desired resource.

Due to their nature, path parameters must be “URL encoded” to remove special characters which would make the URL invalid. While Swagger does automatically URL encode all Path parameter prior to using it, the architecture employed in SmartConnector requires **double URL encoding**. This means that even when using Swagger, you will be required to encode the path parameter once manually. Fortunately, free tools or websites like <http://meyerweb.com/eric/tools/dencoder/> are available to simplify this.

For example, to retrieve a single Container instance by its *Id* parameter, the route [GET /Containers/{id}](#) would be used. Taking the value returned from [GET /Root](#) (Figure 9), URL encoding it, and entering it into the Swagger input page we should see something like Figure 11.



GET /Containers/{id} Returns the Container with the Id provided.

Response Class (Status 200)
Action successful.

Model Example Value

```
{
  "Name": "string",
  "Description": "string",
  "Type": "Folder",
  "ParentId": "string",
  "Id": "string"
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="00%2FServer%201"/>	Id of the Container requested. Must be double URL encoded.	path	string

Figure 11: Swagger – GET /Containers/{id}

As expected, the response shown in Figure 12 is identical in content to that shown in Figure 9. This is because while we have employed two different resource actions, we are still requesting the “root” *Container* in both cases. In the former request, we simply didn’t know what *Id* to use. We did in the later request.

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	00%2FServer%201	Id of the Container requested. Must be double URL encoded.	path	string

[Hide Response](#)

Curl

```
curl -X GET --header 'Accept: application/json' --header 'Authorization: Bearer AQAAANCmnd8BFdERjHoAwE_C1-sBAAAAVW1jLgG33keEiTtmEv'
```

Request URL

```
http://127.0.0.1:8084/Containers/00%252FServer%25201
```

Response Body

```
{
  "Name": "Server 1",
  "Description": "",
  "Type": 1,
  "ParentId": null,
  "Id": "00/Server 1"
}
```

Response Code

```
200
```

Response Headers

```
{
  "date": "Fri, 27 Jan 2017 17:59:28 GMT",
  "server": "Microsoft-HTTPAPI/2.0",
  "content-length": "80",
  "content-type": "application/json; charset=utf-8"
}
```

Figure 12: Swagger – GET /Containers/{id} Response (JSON)

6.1.3.2 Query Parameters

Query parameters, or [query string](#) when multiple parameters are present, are similar in nature to that of Path Parameters in that they are part of the URL itself. What differentiates Query Parameters is that they are not technically part of the resource action. Instead, Query Parameters are added after the resource and action and are used to refine the request.

Like Path Parameters, Query Parameters must be URL encoded prior to use. For our purposes, only a single URL encoding is required and since Swagger does this for us automatically, no special attention is required.

For example, to retrieve all children for a *Container*, the route [GET /Containers/{id}/Children](#) would be used. Using the *Id* value returned from [GET /Root](#) (Figure 9), URL encoding it, and entering it into the Swagger page we can request all children. If we only are interested in children that are also of type *Container*, we select a “type” value. Clicking “Try it now!” we see the result shown in Figure 13.

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="00%2FServer%201"/>	Id of the Container whose children are to be returned. Must be double URL encoded.	path	string
type	<input type="text" value="Container"/>	Return only those children which are of this type. All returned by default.	query	string

[Hide Response](#)

Curl

```
curl -X GET --header 'Accept: application/json' --header 'Authorization: Bearer AQAAANCMnd8BFdERjHoAwE_C1-sBAAAaVW1jLgG33keEiITmEv'
```

Request URL

```
http://127.0.0.1:8084/Containers/00%252FServer%25201/Children?type=Container
```

Response Body

```
[
  {
    "Name": "Servers",
    "Description": "",
    "Type": 0,
    "ParentId": "00/Server 1",
    "Id": "00/Server 1/Servers"
  },
  {
    "Name": "SmartConnector",
    "Description": "",
    "Type": 0,
    "ParentId": "00/Server 1",
    "Id": "00/Server 1/SmartConnector"
  }
]
```

Response Code

```
200
```

Response Headers

```
{
  "date": "Fri, 27 Jan 2017 18:16:30 GMT",
  "server": "Microsoft-HTTPAPI/2.0",
  "content-length": "209",
  "x-total-count": "2",
  "content-type": "application/json; charset=utf-8"
}
```

Figure 13: Swagger – GET /Containers/{id}/Children Response (JSON)

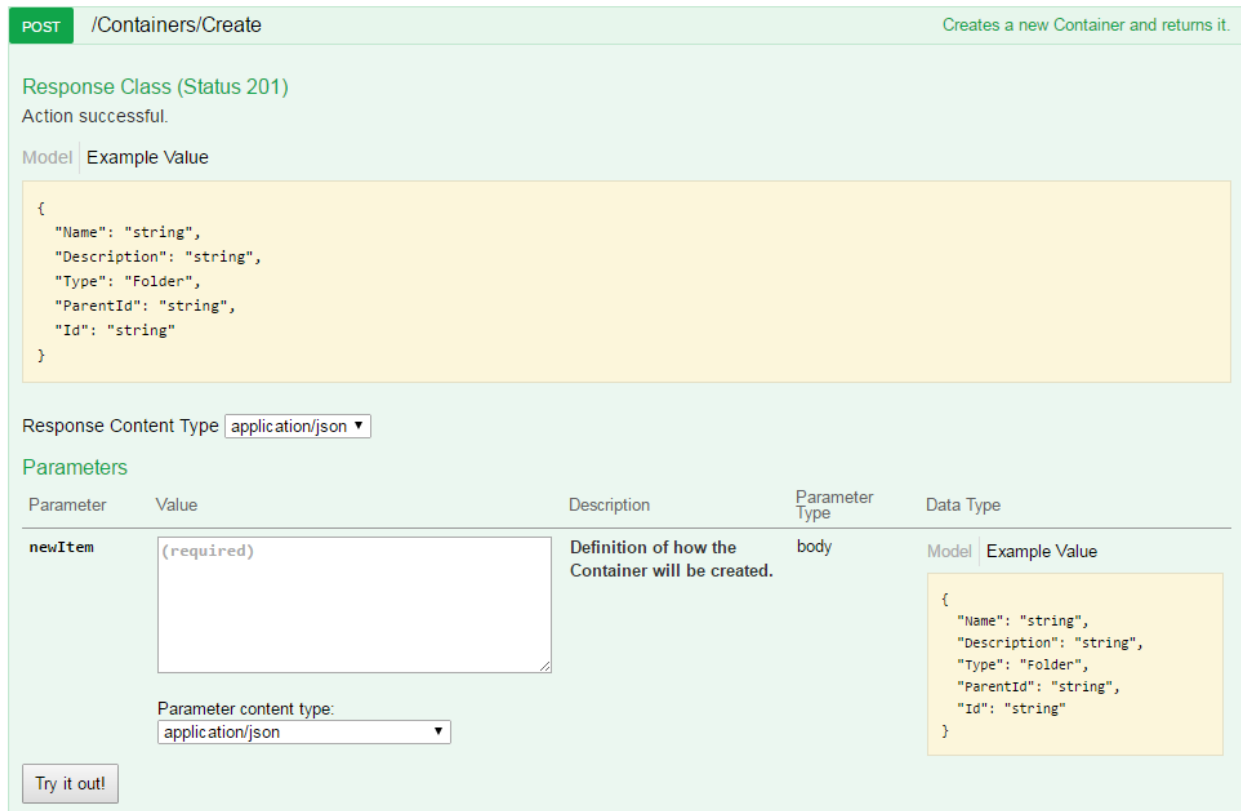
Note the “type=Container” (the numeric or enumeration can be used interchangeably) in the Request URL after the “Children” action. This is how Query Parameters are added to the URL. The first query parameter is preceded by a “?” character while each subsequent query parameter will be preceded with an “&” character.

6.1.3.3 Body Parameters

The final parameter type is that of Body Parameters. Body Parameters are only used in POST, PUT, and PATCH actions.

Since Body Parameters are not part of the URL they do not need to be URL encoded. They do however need to be property formatted depending on the “Parameter content type” option selected.

For example, if we want to add a new Container, we would use the [POST /Containers/Create](#) action as shown in Figure 14.



POST /Containers/Create Creates a new Container and returns it.

Response Class (Status 201)
Action successful.

Model | Example Value

```
{
  "Name": "string",
  "Description": "string",
  "Type": "Folder",
  "ParentId": "string",
  "Id": "string"
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
newItem	(required)	Definition of how the Container will be created.	body	Model Example Value

Parameter content type:

Figure 14: Swagger – POST /Containers/Create

If we choose “application/json” as the “Parameter content type” then the content of “newItem” must be valid JSON. Luckily Swagger assists in this regard. If you click anywhere in the “Example Value”, Swagger will copy the format of the expected content to newItem. As a safeguard, Swagger only does this when there is no content in the target input field. After that, you can quickly edit the markup and submit the request as shown in Figure 15.

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type				
newItem	<pre>{ "Name": "New Container", "Description": "", "Type": "Folder", "ParentId": null, "Id": "MyNewContainer ID" }</pre>	Definition of how the Container will be created.	body	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "Name": "string", "Description": "string", "Type": "Folder", "ParentId": "string", "Id": "string" }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "Name": "string", "Description": "string", "Type": "Folder", "ParentId": "string", "Id": "string" }</pre>
Model	Example Value							
	<pre>{ "Name": "string", "Description": "string", "Type": "Folder", "ParentId": "string", "Id": "string" }</pre>							

Parameter content type:

[Try it out!](#) [Hide Response](#)

Curl

```
curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' --header 'Authorization: Bearer AQAAAAAC'
  "Name": "New Container", \
  "Description": "", \
  "Type": "Folder", \
  "ParentId": null, \
  "Id": "MyNewContainer ID" \
}' 'http://127.0.0.1:8083/Containers/Create'
```

Request URL

`http://127.0.0.1:8083/Containers/Create`

Response Body

```
{
  "Name": "New Container",
  "Description": "",
  "Type": 0,
  "ParentId": "ContainerItem0",
  "Id": "MyNewContainer ID"
}
```

Response Code

201

Response Headers

```
{
  "date": "Fri, 27 Jan 2017 18:42:02 GMT",
  "server": "Microsoft-HTTPAPI/2.0",
  "content-length": "103",
  "content-type": "application/json; charset=utf-8"
}
```

Figure 15: Swagger – POST /Containers/Create Response (JSON)

If XML is your thing, the request would look like Figure 16.

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
newItem	<pre><?xml version="1.0"?> <NewContainerModel> <Name>string</Name> <Description>string</Description> <Type>Folder</Type> <ParentId>string</ParentId> <Id>string</Id> </NewContainerModel></pre>	Definition of how the Container will be created.	body	Model

Parameter content type:

[Hide Response](#)

```
<?xml version="1.0"?>
<NewContainerModel>
  <Name>string</Name>
  <Description>string</Description>
  <Type>Folder</Type>
  <ParentId>string</ParentId>
  <Id>string</Id>
</NewContainerModel>
```

Figure 16: Swagger – POST /Containers/Create (XML parameter)

When the action is a PUT or POST on a scalar value, the “application-www-form-urlencoded” option can also be used. The format used here is slightly different as a preceding “=” character is required.

For example, to change the Type of the Container added above we would use the [PUT /Containers/Type](#) action as shown in Figure 17.

PUT /Containers/{id}/Type Modifies the Type property for the Container with the Id provided to the new value supplied. Returns the modified Container.

Response Class (Status 200)
Action successful.

Model	Example Value
	<pre>{ "Name": "string", "Description": "string", "Type": "Folder", "ParentId": "string", "Id": "string" }</pre>

Response Content Type

Parameters

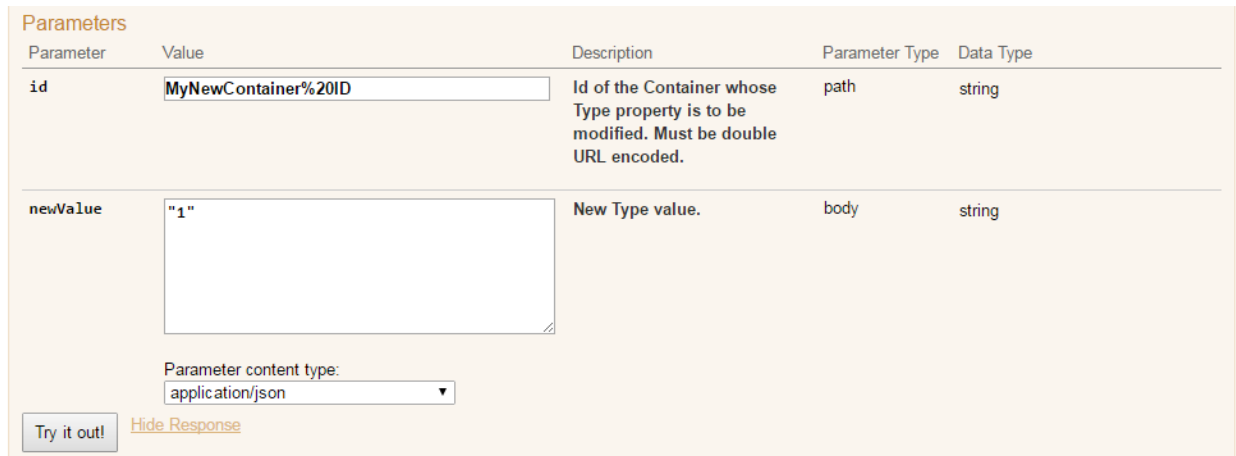
Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="MyNewContainer%20ID"/>	Id of the Container whose Type property is to be modified. Must be double URL encoded.	path	string
newValue	<input type="text" value="=1"/>	New Type value.	body	string

Parameter content type:

[Hide Response](#)

Figure 17: Swagger – PUT /Containers/Type (form-urlencoded parameter)

Or with a JSON parameter as shown in Figure 18.



Parameter	Value	Description	Parameter Type	Data Type
id	MyNewContainer%20ID	Id of the Container whose Type property is to be modified. Must be double URL encoded.	path	string
newValue	"1"	New Type value.	body	string

Parameter content type: application/json

Try it out! [Hide Response](#)

Figure 18: Swagger – PUT /Containers/Type (JSON parameter)

6.2 Metadata and Client Code Generators

SmartConnector serves a Swagger metadata feed for every REST endpoint provisioned. It is this metadata feed that is used to drive the considerable aftermarket ecosystem of client side and test tool code generators.

While SmartConnector makes no recommendations on the best language or code generator to use when consuming SmartConnector REST endpoints, a good place to start is at the Swagger commercial tools site found here: <http://swagger.io/commercial-tools/>.

7 Security Considerations

Security Considerations and recommendations are covered in Section 12 of the *SmartConnector Installation and Configuration Guide*.