SIEMENS

Web Service Interface
Version 2.1

Configuration

A6V10438036_en_a_21
2015-07-15

Siemens Industry, Inc.
Building Technologies

Copyright Notice

Notice

To the Reader

Your feedback is important to us. If you have comments about this manual, please submit them to: SBT_technical.editor.us.sbt@siemens.com

Credits

Desigo, Desigo CC, Cerberus DMS, APOGEE, XLS FireFinder, and Sinteso are registered trademarks of Siemens Industry, Inc.

Other product or company names mentioned herein may be the trademarks of their respective owners.

Edition: 2015-07-15

Document ID: A6V10438036_en_a_21

Table of Contents

About This Document

Purpose

This document describes the workflows between the outside world and the Management System through a Web Service Interface (Web API). It describes how the interface is to be accessed and what data in what format are exchanged.

Scope

This document applies to the system version 2.1.

The document provides a description of the public API which can be used to access data and the functionality provided by the system platform.

Target Audience

- Testers of the public API
- Developers/Clients accessing the public API

Liability Disclaimer

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcome.

Product Security Disclaimer

Siemens products and solutions provide IT-specific security functions to ensure the secure operation of building comfort, fire safety, security management and physical security systems. The security functions on these products and solutions are important components of a comprehensive security concept.

However, it is necessary to implement and maintain a comprehensive, state-of-the-art security concept that is customized to individual security needs. Such a security concept may result in additional site-specific preventive action to ensure that the building comfort, fire safety, security management or physical security systems for your site are operated in a secure manner. These measures may include, but are not limited to, separating networks, physically protecting system components, user awareness programs, in-depth security, and so on.

For additional information on building technology security and our offerings, contact your Siemens sales or project department. We strongly recommend signing up for our security advisories, which provide information on the latest security threats, patches and other mitigation measures.

http://www.siemens.com/innovation/en/technology-focus/siemens-cert/cert-security-advisories.htm

Document Conventions

The following table lists conventions to help you use this document in a quick and efficient manner.

Convention	Examples
Numbered Lists (1, 2, 3…) indicate a procedure with sequential steps.	**1.** Turn OFF power to the field panel. **2.** Turn ON power to the field panel. **3.** Open the panel.
One-step procedures are indicated by a bullet point.	● Expand the **Event List**.
Conditions that you must complete or must be met before beginning a procedure are designated with a ▷. Intermediate results (what will happen following the execution of a procedure step), are designated with an indented ⇨. Results, after completing a procedure, are designated with a ⇨.	▷ The report you want to print is open. **1.** Click **Print**  . ⇨ The **Print** dialog box displays. **2.** Select the printer and click **Print**. ⇨ The print confirmation displays.
Bold font indicates something you should type or select, or when a dialog box or window is specified.	Type **F** for field panels. Click **OK** to save changes and close the dialog box. The **Create a New Project** dialog box displays.
Menu paths in procedures are indicated in **bold**.	Select **File > Text, Copy > Group**, which means from the **File** menu, select **Text**, **Copy** and then **Group**.
File paths containing placeholders display the placeholders in *italics* enclosed in square brackets.	[*installation drive*:]\[*installation folder*]\[*project*]\...
Error and system messages are displayed in `Courier New` font.	The message `Report Definition successfully renamed` displays in the status bar.
Italics are used to emphasize new or important terms.	The reaction processor continuously executes a user-defined set of instructions called the *control program*.
	This symbol signifies a Note. Notes provide additional information or helpful hints.
Cross references to other information in printed material are indicated with an arrow and the page number, enclosed in brackets: [→ 92]	For more information on creating flowcharts, see Flowcharts [→ 92].

Getting Help

For more information about our products, contact your local Siemens representative.

Safety Messages According ANSI Z535.6

The following examples show the ANSI standard safety messages used in this document to draw the reader's attention to important information.

ANSI distinguishes between *personal injury* safety messages and *property damage* warning messages.

The personal injury safety messages have safety alert symbols and the following alert level labels: DANGER!, WARNING!, CAUTION!

The label for property damage messages is: NOTICE.

Examples:

	*NOTICE*
	Property Damage Warning Message Equipment damage or loss of data may occur if you do not follow a procedure or instruction as specified.

	⚠ CAUTION
	Caution Safety Message Minor or moderate injury may occur if you do not follow a procedure or instruction as specified.

	⚠ WARNING
	Warning Safety Message Personal injury or property damage may occur if you do not follow a procedure as specified.

	⚠ DANGER
	Danger Safety Message Electric shock, death, or severe property damage may occur if you do not perform a procedure as specified.

Document Revision History

Document Identification

The document ID is structured as follows:
ID_Language(COUNTRY)_ModificationIndex_ProductVersionIndex
Example: A6Vnnnnnnnn_en_a_02

Document Revision History.		
Modification Index	**Edition Date**	**Brief Description**
a	2015-07-15	Market Release Edition

1 Introduction

1.1 References

1.1.1 Links

Ref.	Abbreviation	Link Description
[1]	HAL	http://stateless.co/hal_specification.html
[2]	SignalR	https://github.com/SignalR/SignalR/wiki
[3]	Base64url	http://www.ietf.org/rfc/rfc4648.txt
[4]	OAuth	https://tools.ietf.org/html/rfc6749
[5]	Langauge-Tags	http://tools.ietf.org/html/rfc5646
[6]	JSON:	http://www.ietf.org/rfc/rfc4627.txt
[7]	REST	http://en.wikipedia.org/wiki/Representational_state_transfer
[8]	URI	https://tools.ietf.org/html/rfc3986
[9]	EBNF	http://en.wikipedia.org/wiki/Extended_Backus%E2%80%93Naur_Form
[10]	Media-Types	http://www.iana.org/assignments/media-types/media-types.xhtml
[11]	Swagger	http://swagger.io

1.1.2 Documents

Ref.	Name	Version
[20]	Regional Product Customization Guide	

1.1.3 Online Documentation

Web Service Interface also provides an interactive online documentation (web page). The documentation is based on Swagger [11] and is a runtime representation of the API. It requires a running Web Service Interface and can be accessed by the following URL:

```
http(s)://[hostname]:[port]/swagger/ui/index
```

Example

```
https://myWebServiceHost:8443/swagger/ui/index
```

NOTE:
If you are using Internet Explorer make sure Compatibility View is turned off.

1.2 Definitions, Acronyms and Abbreviations, Conventions

1.2.1 Glossary

Terms Used in Web Service Interface	
Term	**Description**
HAL	Simple format that gives a consistent and easy way to hyperlink between resources in an API.
HTTP	Hypertext transfer protocol: http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol
JSON	JavaScript Object Notation: http://en.wikipedia.org/wiki/JSON
OAuth 2.0	Open standard to authorization. OAuth provides client applications a 'secure delegated access' to server resources on behalf of a resource owner.
REST	Representational state transfer: http://en.wikipedia.org/wiki/Representational_state_transfer
Real-time push	Push notification from a server (our Web Service in this case) to a client; used for notification of changes of values as long as client application is running.
Reverse Proxy	Retrieves resources on behalf of a client from a server. These resources are then returned to the client as though they originated from the server itself.
SignalR	Library for ASP.NET. SignalR allows bi-directional communication between server and client.
Wake-up push	Push notification to a mobile phone; notifies a user about an event in the system; takes advantage of third-party notification providers (Apple, Google, …).

1.2.2 Conventions in this Document

The product is developed for different regions and known under the following brands:

- Desigo CC
- Desigo DMS
- Cerberus DMS

In this document, we therefore use the neutral term *System* for each of these brands.

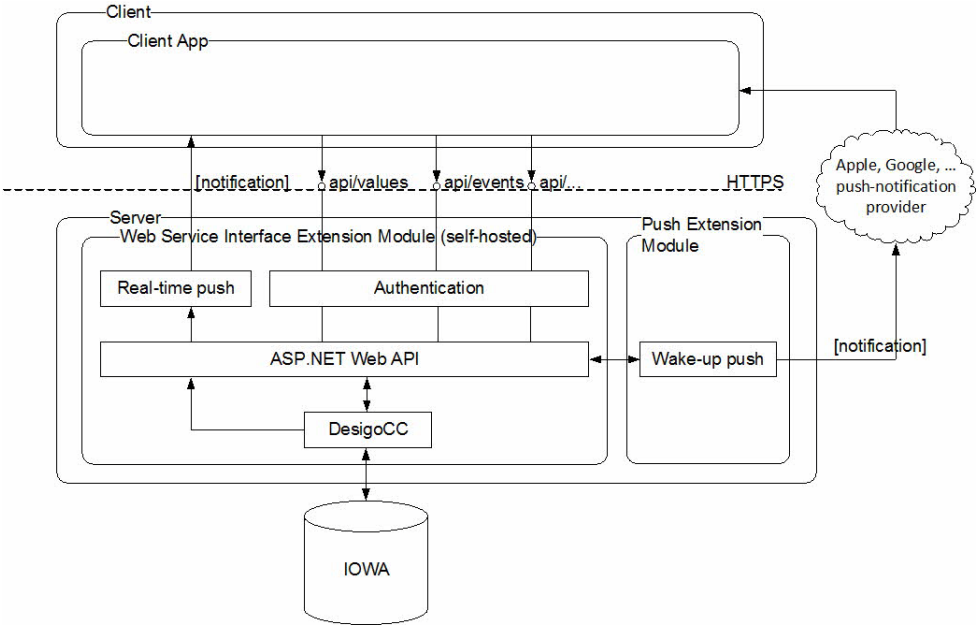
1.3 Open Issues in this Document Version

At the effective date of the current document version, the following open issues are identified:

- none

2 System Architecture

The following diagram shows the architecture for the Web Service Interface API.



NOTE:
Wake-up push notifications are part of a future extension module.

2.1 System Limits

The following limits apply when using the Web Service Interface:

- Number of concurrent sessions: 100
- Delay for notifications: <1 second
- 250’000 value updates / day
- 10’000 value subscriptions
- 50’000 event notifications / day
- 50’000 event-counter notifications / day
- 1’000’000 trends / min

3 Using the Web Service Interface

3.1 Communication

The API can be accessed over HTTP or HTTPS. In a production environment it's highly recommended to use HTTPS because otherwise primary credentials are transmitted in plaintext.

3.1.1 Authentication

For authentication we use OAuth 2.0 Resource Owner Password Credentials Grant [4] [→ 12]. In order to access a resource of the Web Service Interface the Client needs a valid access token and a valid session within the system. In case no token is presented or the token is invalid (e.g. expired) or the system session is available the user will get a status code 401 (`Unauthorized`) and needs to request an access token. The access token can be requested through a dedicated resource (/token) by presenting the primary credentials. In case the credentials are provided, the system creates a new session and returns an access token.

1. Client tries to access a resource.
2. Server responses with status code 401 (`Unauthorized`).
   ⇨ Client asks the end-user for credentials.
3. Client sends credentials to a dedicated resource.
4. Server returns an access token, the system creates an internal session.
5. Client again tries to access a resource and includes the access token in an authorization header.
6. Server returns content with status code 200 (`Succeeded`).

!	*NOTICE*
	Session alive The Client either needs to subscribe for notifications (e.g. change of values) or needs to access the API at least once every 10 minutes otherwise its system session expires. The API provides a dedicated resource (/heartbeat) just for the sake of keeping a session alive.

3.1.2 Client Certificate Authentication

Optionally, a client can authenticate to the server by providing a client certificate. The Web Service Interface expects the client certificate to be provided in a header called `X-ARR-ClientCert`. In case the interface is accessed through a reverse proxy (almost always the case in a production environment), the reverse proxy asks the client for an optional client certificate. In this case the reverse proxy passes the client certificate to the backend server as HTTP header with the default header configured as `X-ARR-ClientCert`.

3.1.3 Subscriptions (Push Notifications with SignalR)

Optionally, for specified services the API provides not only data on request (pull) but also in case of an event determined by the respective service (push).

Any Client subscribing for any push notifications needs to support SignalR (see [2] [→ 12]). The Client first needs to connect to a dedicated SignalR hub and can then subscribe for notifications by providing a connection ID which it gets after a connection between Client and server is established. The Client then needs to implement a function (see signature in respective service chapters) which will be called from the server in case a notification is due.

Example

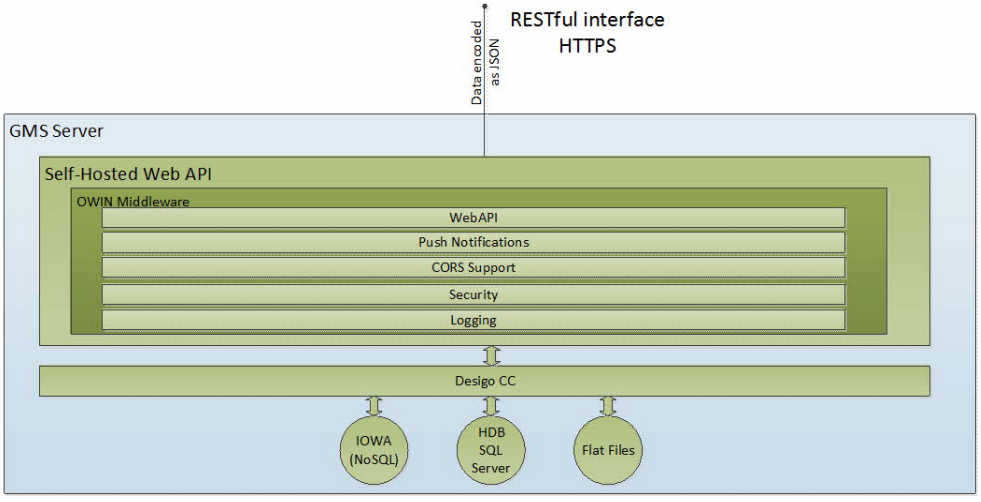
```
//eventsHub is name of our hub on the server side
var hub = $.connection.eventsHub;

//register notifyEvents as a method on the client side
hub.client.notifyEvents = function (events) {
    //do something in case a notification arrives
};

//start hub
$.connection.hub.start().done(function () {
    //do other stuff
});
```

3.2 Supported Client Environments

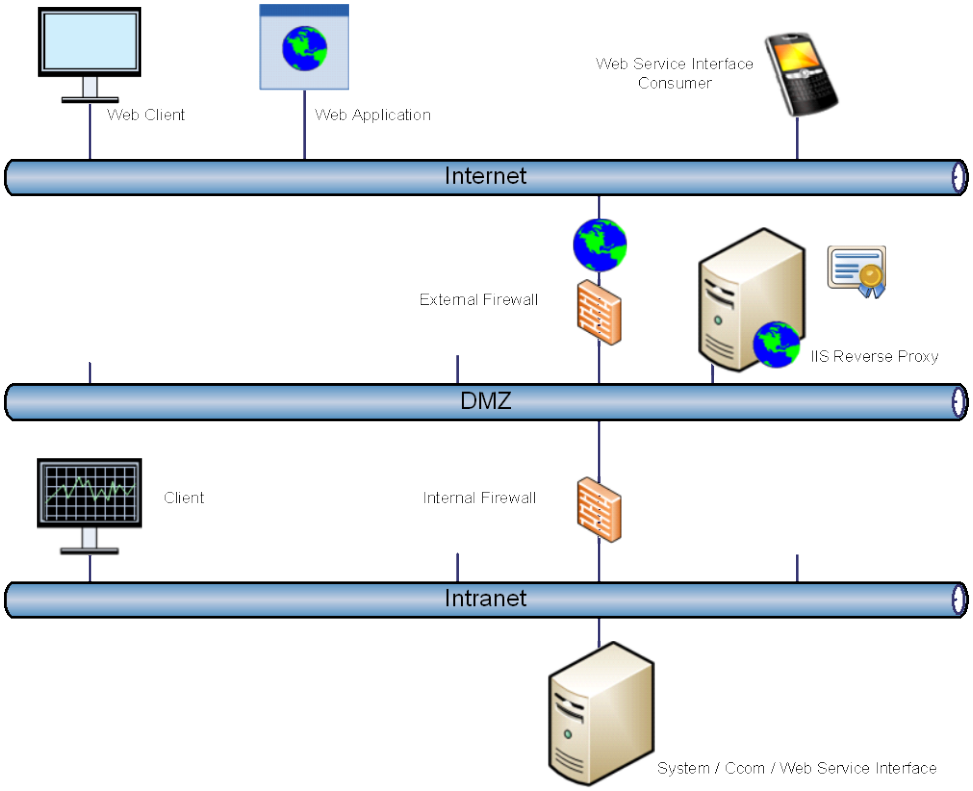
There are no Client dependencies except SignalR Client library (see [2] [→ 12] for available Client libraries) for Clients interested in push notifications.

3.3 Deployment

The Web Service Interface will be deployed as a self-hosted component. It's an executable and can be commanded as any other IOWA manager [12] [→ 12].



In a production environment the Web Service Interface (and System) will most likely only be accessible through a reverse proxy.



Network Topology

3.4 Configuring Web Service Interface in a Project

NOTE:
A user has the same user rights as defined in the system. Therfore, you can check the user rights in advanced in the system bevroe using the Web Service Interface.

3.4.1 Stopping a Project

▷ At least one project is available under the **Projects** node and is started.

1. In the System Management Console tree, select the **Projects** node.
2. In the **Projects** tree, select the project that you want to stop.
3. Click the **Project Settings** tab, if not yet selected.
   ⇨ **Project Settings** displays.
4. In the **Project Settings** toolbar, click **Stop** .
   ⇨ A confirmation message displays.
5. Click **OK** to stop the started project.

3.4.2 Showing Settings

▷ The Extension module **Connectivity > Web Service Interface** is installed.

▷ At least one project is available under the **Projects** node and is started.

1. In the System Management Console tree, select the **Projects** node.
2. In the **Projects** tree, select the project.
3. Click the **WSI Settings** tab.
   ⇨ **WSI Settings** displays and shows the used protocol type and port number.

NOTE:
If you want to change the settings, you must stop the project first.

3.4.3 Defining Protocol Type and Port

▷ The *Web Service Interface* extension module installed for the selected project.

▷ The project is stopped.

▷ IIS is installed according to the settings in the Installation Manual chapter 4.

1. Click **Edit** .
2. Click **Next** .
   ⇨ The **Web Service Interface Settings** expander displays.
3. Select the protocol:
   - **Unsecured**
     - Under **Port**, enter the port number (Default: 8080).
   - **Secured**
     – Under **Port**, enter the port number (Default: 8443).
     – Under **Host Certificate**, click **Browse**.
     – Select the appropriate certificate and click **OK**.
     NOTE: Additional information on certificates is available in the SMC Help.
4. Click **Save Project** .
   ⇨ The protocol settings are saved.
5. Click **Start Project** .

NOTE:
HTTP is insecure and is vulnerable to man-in-the-middle and eavesdropping attacks, which can let attackers gain access to sensitive information.

Many organizations require certificates from third-party authorities instead of self-signed certificates to remain compliant with current regulations.

3.4.4 Creating a WSI on Server

▷ At least one Web site is created and available under the **Websites** root node in the System Management Console tree.

1. In the System Management Console tree, select the **Websites** node.
2. In the **Websites** tree, select the **Website**.
3. Click the **Management** tab.
   ⇨ The **Details** expander displays.
4. In the **Website** toolbar, click **Web Service Interface** .
   ⇨ The **Web Application Details** expander displays.
5. In the **Web Application Details** expander, provide the Web application details as follows:
   - From the **Project Name** drop-down list, select a project that you want to link to the Web application.
   - In the **Name** field, type a unique name of the Web application.
   - Browse for the **Path** to save the Web application on the disk. The default path is [*installation drive*:]\[*installation folder*]\[WebSites]\[*Web site Name*].
   - Browse for a web application **User** using the Select User dialog box. You can select a different user than the one you selected during Web site creation.
     NOTE: The Web application must be a member of the IIS_IUSRS group. If you select a user that is not a member of the IIS_IUSRS group, SMC prompts you to add it.
   - Enter the valid **Password** for the Web application user.
6. Click **Save** .
   ⇨ A confirm message displays.
   ⇨ Click **OK** to initiate the Web application creation (or **Cancel** to abort).

⇨ If confirmed, the data is validated on successful creation.

⇨ The **Web Service Interface Settings** expanders displays the defined settings.

3.4.5 Starting a Project

▷ The project is stopped.

1. In the System Management Console tree, select the **Projects** node.
2. In the **Projects** tree, select the project that you want to start.
3. Click the **Project Settings** tab, if not yet selected.
   ⇨ **Project Settings** displays.
4. In the **Project Settings** toolbar, click **Start Project** .
   ⇨ A confirmation message displays.
5. Click **OK** to start the stopped project

4 Conventions

4.1 Case Sensitivity

Unless otherwise specified the system is case sensitive. This affects the API in terms of filters and search results.

4.2 Number Format

The API uses an invariant culture for the number formatting regardless of the logged in user. The decimal mark is a period.

4.3 Date/Time Format

Date and time representations are supposed to be according to ISO-8601.

NOTE:
Unless otherwise noted, the Zulu time zone is expected.

Example

```
2011-12-19T15:28:46.493Z
```

4.4 Terminology

Terminology	
Term	**Description**
Name[1]	Literal; supposedly unreadable; usually not displayed in a UI.
Descriptor[1]	Localized string in a readable form; supposed to be displayed to an end- user.
Designation[1]	Unique address of a node; supposedly unreadable; usually not displayed in a UI.
Location[1]	Localized, fully expanded path of a node; supposed to be displayed to an end-user.
ObjectId	Identifies an object in the system.
PropertyId	Identifies a property and its object in the system.
AttributeId	Identifies an attribute, its property and its object in the system.
ObejctOrPropertyId	An object ID or a property ID is expected.
PropertyName	Name of a property.

[1] see also Naming [➙ 69]

5 General API Specification

The REST API provides programmatic access to read, write and command the system data and functionality. The API identifies users using OAuth [4] [➞ 12]; responses are available in JSON [6] [➞ 12].

5.1 HTTP Methods

Following REST principles, one can call the following HTTP methods on the API resources: GET, POST, PUT, and DELETE.

Method (HTTP Verb)	Description
GET	Retrieves a list of resources, or a specific resource.
POST	Creates a resource, passes in a parameter, or performs an action on a resource.
PUT	Updates an existing resource.
DELETE	Deletes an existing resource or subscription.

5.2 Resources

The Web Service Interface API follows REST principles and therefore exposes data and functionality as resources [7] [➞ 12].

Every resource is represented as an URI (see Appendix A in [8] [➞ 12]). As a summary a resource looks like the following:

```
scheme://domain:port/path?query
```

Part	Description
scheme	Usually HTTPS; HTTP should only be used for testing
domain	Hostname of host where manager is running
port	Usually 8080 for HTTP and 8443 for HTTPS
path	Starts with api followed by the service-identifier and optional sub paths (e.g. api/languages/en-us); see sections below
query	Optional; depends on resource (see sections below)

Example

```
https://ch1w80012:8443/api/events?sorting=1
```

NOTE:
In a production environment, the API runs most likely behind a reverse proxy. In this case, the URI might look differently and might hide some information from the Client.

Example

```
https://www.example.com/example/api/events?sorting=1
```

Therefore the general format of a URL is:

```
{entry-point}/api{resource-address}
```

Where **entry-point** is http://[hostname]:[port] by default or dependent on the deployment, and **resource-address** in EBNF is:

```
resource-address = path [ "?" query ]
path = /root-resource ¦ path "/" sub-resource
query = parameter { "&" parameter }
parameter = key [ "=" value ]
```

5.3 Status Codes

5.3.1 HTTP Status Codes

Status Code	Meaning	Description
200	OK	Request processed. **NOTE**: Although the request has been processed as a whole, some of the sub results could still be invalid. See Error Code [→ 22] to check sub-results.
400	Bad Request	Request rejected by application.
401	Unauthorized	Invalid or expired token presented. User needs to login.
404	Not Found	Resource couldn't be found for the provided URL and query parameters.

5.3.2 Error Code

Code	Description
0	No error occurred.

5.4 Encoding

Data is encoded in UTF-8.

5.5 Array in Query Parameter

In case a query parameter represents an array, the array is passed as a JSON array [6] [→ 12].

Example

```
someurl?Ids=[1,2,3]
```

6 Services

6.1 Token Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/token	x (Logout only)	

6.1.1 Logging in

Logging in a User by Providing Username and Password			
Resource Path	/token		
HTTP Method	POST		
Content Type	application/x-www-form-urlencoded		
Body	**Property**	**Content**	**Description**
	grant_type	password	grant_type=password is the only supported login flow.
	username	[*username*]	Username provided by user.
	password	[*password*]	Password provided by user.
Response	HTTP 200 OK A Login object (see Token Service Login [→ 62])		
Examples	**Request:** `POST /NBWSITest/NorthboundWebApiClient/api/token` **Post Data:** `grant_type=password&username=testuser&password=123` **Response:** `{` `"access_token": "AQAAANCMnd8BFdERjHoAwE_Cl-sBAAAAF_QcuV15NE-4HzTEsvL8egAAAAACAAAAAAADZgAAwAAAABAAAAA7E1xxMxM1h3TMh6J3E2L8AAAAAASAAACgAAAAEAAAALKEbFDxydBT5lR6CwWASoYgAQAA1Wd3v8tiUqBQBStWnhvVt1dtR83TeVPhKytj9dR2MaeSEEuhpQCuvCtMZv_uJOvwF9fUpAHMN11xhir_iIci6Fm1iocvHOMzlg19QSgClZdbFwWq--wwHjHugLMCCKYZop52PR8a5qoHaj_BkvBaX2b8k9FkYGIk_gUsoJvbubg3-U4Gyrc1kEiRItg2Y9EiIYlv2MD1KN_VsEa4rXwygnX9J69PIJrvzsleLoizHodOdXXF1wnpcG_BRaZOgEb6imSIqXqu1-TyATYHCGRfmdjnLuX_V2L-oXNCcdk3eFnUFSjmCZQJIZY2NJaarTNLrwsOjU5J7MAmQC2A5l8tGVEoqFIV-NP6qW6lq2yyx51eIIcUSicZ4nrwpKMvuGHEFAAAAHDXVd9LDvd8cGQpM33pTcwkWl-u",` `"token_type": "bearer",` `"expires_in": 2591999` `}`		

6.1.2 Logging out

Logging out a User	
Resource Path	/token
HTTP Method	DELETE
Response	HTTP 200 OK

Examples	**Request:**
	`DELETE /NBWSITest/NorthboundWebApiClient/api/token`
	Response:
	`HTTP 200 OK`

6.2 Heartbeat Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/heartbeat	x	

6.2.1 Extending the Lifetime of a Session and its Bearer Token

If a Client logs in, a session is created and bound to the bearer token that the Client receives as part of a successful login request. The session and its bearer token expire after some time of inactivity (default: 10 minutes). Inactivity is detected if no subscription is alive and no protected (authenticated) call is made within this timeout.

The **/heartbeat** resource can be called to artificially signal activity and to reset the timeout.

Lifetime of a Session.	
Resource Path	/heartbeat
HTTP Method	POST
Response	HTTP 200 OK

Examples	**Request:**
	`POST /NBWSITest/NorthboundWebApiClient/api/heartbeat`
	Response:
	`HTTP 200 OK`

6.3 Event Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/events	x	
/eventssubscriptions	x	x

Events depend on the event type (see [20]). The current event type can be read with the following object ID: ManagementView_System_Settings.EventSchema.

6.3.1 Retrieving a List of Events

Resource Path	/events		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	sorting	int	See EventSorting [→ 27]
	page	int	
	size	int	
	ids	Array of string	
	caseSensitive	bool	
	categories	Array of int	
	cause	string	
	creationTime	int	See EventTimeFilter [→ 26]
	creationTimeFrom	DateTime	
	creationTimeTo	DateTime	
	disciplines	Array of int	
	states	Array of string	See EventState [→ 62]
Response	HTTP 200 OK A Page<Events, Event>-object (see Page<Name, Type>) [→ 58]		

Examples	**Request:**
	`GET /api/events?sorting=0`
	Response:
	`{`
	`  "Events": [`
	`  {`
	`  "Id": "U3lzdGVtMTpHbXNEZXZpY2VfMV8xXzM3NzQ4NzM3LkFsYXJtLk9mZk5vcm1hbDpfYWxlcnRfaGRsLjIuX3ZhbHVl~6356356919360000000~6146",`
	`  "EventId": 3121,`
	`  "CategoryId": 3,`
	`  "CategoryDescriptor": "Security",`
	`  "State": "Unprocessed",`
	`  "Cause": "Low Limit (35.00 °F)",`
	`  "SrcPropertyId": "System1:GmsDevice_1_1_37748737.Alarm.OffNormal:_alert_hdl.2._value",`
	`  "SrcObservedPropertyId": "System1:GmsDevice_1_1_1.Present_Value:_original.._value",`
	`  "SrcState": "Quiet",`
	`  "SrcSystemId": 1,`
	`  "SrcViewName": "ManagementView",`
	`  "SrcViewDescriptor": "Management View",`
	`  "SrcDesignation": "ManagementView.FieldNetworks.MyBAC0.Hardware.1/1-3457.1/0-Local_IO.1/1-EE_1",`
	`  "SrcLocation": "Project.Field Networks.My BAC0.Hardware.Simulator Device 1.Local_IO.Event Enrollment 1",`
	`  "SrcName": "1/1-EE_1",`
	`  "SrcDescriptor": "Event Enrollment 1",`
	`  "SrcDisciplineId": 50,`
	`  "SrcDisciplineDescriptor": "Building Automation",`
	`  "CreationTime": "2015-04-02T08:59:53.6Z",`
	`  "Direction": "None",`		

```
      "InfoDescriptor": "BACnet Priority: 40 - BACnet
Notification Class: 1  - Message Text #1 at (Time
10:59:53)",
      "Commands": [
        {
          "Id": "Suspend",
          "EventId":
"U3lzdGVtMTpHbXNEZXZpY2VfMV8xXzM3NzQ4NzM3LkFsYXJtLk9mZk5vcm1hbDpfYWxlcnRfaGRsLjIuX3ZhbHVl~635635691936000000~6146",
          "_links": [
            {
              "Rel": "command",
              "Href":
"api/eventscommands/U3lzdGVtMTpHbXNEZXZpY2VfMV8xXzM3NzQ4NzM3LkFsYXJtLk9mZk5vcm1hbDpfYWxlcnRfaGRsLjIuX3ZhbHVl~6356356
91936000000~6146/Suspend",
              "IsTemplated": false
            }
          ]
        }
      ],
      "_links": [
        {
          "Rel": "category",
          "Href": "api/tables/categories/3",
          "IsTemplated": false
        }
      ]
    }
  ],
  "Total": 3,
  "Page": 1,
  "Size": 200,
  "_links": [
    {
      "Rel": "page",
      "Href": "api/events?page={page}",
      "IsTemplated": true
    }
  ]
}
```


6.3.1.1 EventTimeFilter

creationTime must have one of the following values:

creationTime	Description
0	All events (default, can be omitted)
1	Events generated in the last 15 minutes
2	Events generated in the last 30 minutes
3	Events generated in the last hour
4	Events generated last night
5	Events generated yesterday
6	Events generated today

6.3.1.2 EventSorting

sorting must have one of the following values.

sorting	Description
0	Sort by StateId, CategoryId, and descending CreationTime
1	Sort by StateId, CategoryId, DirectionId, and ascending CreationTime

6.3.2 Creating a Subscription for Events

Resource Path	/sr/eventssubscriptions/:connectionId		
HTTP Method	POST		
Parameters	**Property**	**Type**	**Description**
	connectionId	string	Identifier of subscription **Required**
	sorting	int	See Event Sorting [→ 27]
	ids	Array of strings	
	caseSensitive	bool	
	categories	Array of int	
	cause	string	
	creationTime	int	See EventTimeFilter [→ 26]
	creationTimeFrom	DateTime	
	creationTimeTo	DateTime	
	disciplines	Array of int	
	states	Array of int	
Response	HTTP 200 OK		

Examples	**Request:**
	`POST /api/sr/eventssubscriptions/983a0ade-7e7a-4812-bdf3-ea9ea59a7e9e?sorting=0`
	Response:
	`HTTP/1.1 200 OK`		

6.3.2.1 Push Notification

After subscribing for events, a list of current events will be sent in an initial notification. Afterwards, the Client will get notified about events according to the filter set (if any).

Type	Name (case sensitive)		
Hub	eventsHub		
Function	notifyEvents (data)		
	Property	**Type**	**Description**
	data	Array of Event	See Event [→ 61] Check the flag Deleted for checking whether the event got removed.

6.3.3 Modifying a Subscription for Events

Change an existing subscription for events: the same procedure as Creating a Subscription for Events [→ 27] but use PUT as HTTP method.

6.3.4 Deleting a Subscription for Events

Delete an existing subscription for events: the same procedure as Creating a Subscription for Events [→ 27] but use DELETE as HTTP method.

6.4 EventsCommands Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/eventscommands	x	

6.4.1 Executing a Command on an Event

Resource Path	/eventscommands/:eventId/:commandId		
HTTP Method	POST		
Parameters	**Property**	**Type**	**Description**
	eventId	string	**Required**
	commandId	string	See EventCommandId [→ 27] **Required**
Response	HTTP 200 OK		

Examples	**Request:**
	`POST /api/eventscommands/U3lzdGbHVl~635642600000~6598/ack`
	Response:
	`HTTP/1.1 200 OK`		

6.4.1.1 EventCommandId

:commandId must have one of the following values.

:commandId	Description
select	Signals operator is treating an event
suspend	Signals operator stopped treating an event
ack	Acknowledge event
reset	Reset event
silence	Silence event
unsilenced	Unsilence event
close	Close event

6.5 EventCounter Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/eventcounters	x	
/eventcounterssubscriptions	x	x

6.5.1 Retrieving Event Counters

When retrieving Event counters two options are available with parameter categoryId which is optional. If categoryId is provided then Event counter for that particular categoryId will be returned as EventCounter object. And if categoryId is not provided then Event counters for all the categories in the system will be provided as an EventCounterList object. In both the cases response will be different depending on the request made. For more details see the examples below.

<table>
<tr><td>Resource Path</td><td colspan="3">/eventcounters/:categoryId</td></tr>
<tr><td>HTTP Method</td><td colspan="3">GET</td></tr>
<tr><td rowspan="2">Parameters</td><td>Property</td><td>Type</td><td>Description</td></tr>
<tr><td>categoryId</td><td>int</td><td>Identifier of category</td></tr>
<tr><td>Response</td><td colspan="3">HTTP 200 OK
1. When categoryId is provided: EventCounter object see [➞ 52]
2. When categoryId is not provided: EventCounterList object see [➞ 52]</td></tr>
<tr><td>Examples</td><td colspan="3">Request1: categoryId is provided
GET api/eventcounters/9
Response1:
{
"CategoryId": 9,
"CategoryDescriptor": "Fault",
"TotalCount": 1,
"UnprocessedCount": 1,
"_links": [
{
"Rel": "eventcounter",
"Href": "api/eventcounters/9",
"IsTemplated": false
}
]
}

Request2: categoryId is NOT provided
GET api/eventcounters

Response2:
{
"TotalCounters": 1,
"TotalUnprocessedCounters": 1,
"EventCategoryCounters": [
{
"CategoryId": 1,
"CategoryDescriptor": "Emergency",
"TotalCount": 0,
"UnprocessedCount": 0,
"_links": [
{
"Rel": "eventcounter",
"Href": "api/eventcounters/1",
"IsTemplated": false
}
]
},
{
"CategoryId": 2,</td></tr>
</table>

```
      "CategoryDescriptor": "Life Safety",
      "TotalCount": 0,
      "UnprocessedCount": 0,
      "_links": [
        {
          "Rel": "eventcounter",
          "Href": "api/eventcounters/2",
          "IsTemplated": false
        }
      ]
    },
    {
      "CategoryId": 3,
      "CategoryDescriptor": "Security",
      "TotalCount": 0,
      "UnprocessedCount": 0,
      "_links": [
        {
          "Rel": "eventcounter",
          "Href": "api/eventcounters/3",
          "IsTemplated": false
        }
      ]
    }
  ]
}
```


6.5.2 Creating a Subscription for Event Counters

Resource Path	/sr/eventcounterssubscriptions/:connectionId		
HTTP Method	POST		
Parameters	**Property**	**Type**	**Description**
	connectionId	string	Identifier of subscription **Required**
Response	HTTP 200 OK		

Examples	**Request:**
	`POST api/sr/eventcounterssubscriptions/448b3117-44ef-4d8f-baa2-7274aa3050af`
	Response:
	`HTTP/1.1 200 OK`		

6.5.2.1 Push Notification

After subscribing for event counters, the current event counters will be sent in an initial notification. Afterwards, any change will be notified.

Type	Name (case sensitive)		
Hub	eventCountersHub		
Function	notifyEventCounters(data)		
	Property	**Type**	**Description**
	data	EventCounterList	See EventCounterList [→ 52]

6.5.3 Deleting a Subscription for Event Counters

Delete an existing subscription for event counters: the same procedure as creating event counters subscription [→ 32] but use DELETE as HTTP method.

6.6 System Browser Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/systembrowser	x	

6.6.1 Retrieving a List of Views of a System

Retrieves list of Views in the current system. This interface have two optional query parameters systemId and viewId. If systemId is received as zero or if it is null then current systems id will be taken else received systemId will be taken for further operations. If viewId is provided then only the view, which have viewId same as that of in the re quest will be returned in the response. If viewId is not provided then all the views in the system will be returned in the response.

Resource Path	/systembrowser		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	systemId	uint	ID of a system
	viewId	uint	ID of specific View to be retrieved
Response	HTTP 200 OK An array of View objects see [→ 55]		

Examples	**Request:**
	`GET api/systembrowser?systemId=1&viewId=10`
	Response:
	`[ { "SystemId": 1, "ViewId": 10, "Descriptor": "Application View", "Designation": "System1.ApplicationView", "Name": "ApplicationView", "_links": [ { "Rel": "systembrowser", "Href": "api/systembrowser/1/10/U3lzdGVtMS5BcHBsaWNhdGlvblZpZXc", "IsTemplated": false } ] } ]`		

6.6.2 Retrieving a List of Browser Objects

Retrieve a list of browser objects for a specific view. To retrieve child objects of an object pass Designation of the parent object as node parameter.

<table>
<tr><td>Resource Path</td><td colspan="3">/systembrowser/:systemId/:viewId/:node</td></tr>
<tr><td>HTTP Method</td><td colspan="3">GET</td></tr>
<tr><td rowspan="4">Parameters</td><td>Property</td><td>Type</td><td>Description</td></tr>
<tr><td>systemId</td><td>uint</td><td>ID of a system
Required</td></tr>
<tr><td>viewId</td><td>uint</td><td>ID of a view
Required</td></tr>
<tr><td>node</td><td>string</td><td>Designation of a parent browser object (Node or View)
Required</td></tr>
<tr><td>Response</td><td colspan="3">HTTP 200 OK
An array of BrowserObject objects see [→ 55]</td></tr>
<tr><td>Examples</td><td colspan="3">Request:
GET
api/systembrowser/1/9/U3lzdGVtMS5NYW5hZ2VtZW50VmlldzpNYW5hZ2VtZW50Vmlldw
Response:
<pre>[
  {
    "HasChild": true,
    "SystemId": 1,
    "ViewId": 9,
    "Name": "ManagementSystem",
    "Descriptor": "Management System",
    "Designation":
"System1.ManagementView:ManagementView.ManagementSystem",
    "ObjectId":
"System1:ManagementView_Management_System",
    "Attributes": {
      "DefaultProperty":
"StatusPropagation.AggregatedSummaryStatus",
      "ObjectId": "ManagementView_Management_System",
      "DisciplineDescriptor": "Management System",
      "DisciplineId": 0,
      "SubDisciplineDescriptor": "System Settings",
      "SubDisciplineId": 4,
      "TypeDescriptor": "View Element",
      "TypeId": 8000,
      "SubTypeDescriptor": "System Folder",
      "SubTypeId": 8014,
      "ManagedType": 149
    },
    "Location": "System1.Management
View:Project.Management System",
    "_links": [
      {
        "Rel": "systembrowser",
        "Href":
"api/systembrowser/1/9/U3lzdGVtMS5NYW5hZ2VtZW50VmlldzpNYW5
hZ2VtZW50Vmlldy5NYW5hZ2VtZW50U3lzdGVt",
        "IsTemplated": false</pre></td></tr>
</table>

	<pre>},
	{
	"Rel": "propertyvalues",
	"Href":
	"api/propertyvalues/U3lzdGVtMTpNYW5hZ2VtZW50Vmlld19NYW5hZ2VtZW50X1N5c3RlbQ",
	"IsTemplated": false
	}
	]
	}
	]</pre>

6.6.3 Searching for Browser Objects

Search for browser objects in a provided system.

Resource Path	/systembrowser/:systemId/:viewId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	systemId	uint	ID of a system **Required**
	viewId	uint	ID of the view in which search needs to be performed
	searchString	string	String to be searched for **Required**
	searchOption	int	See SearchOption [→ 37]
	caseSensitive	bool	Only valid for search options 0 or 1
	groupByParent	bool	If true, group search results by parent node
	size	int	Number of browser objects per page
	page	int	Current page number (default: 1)
	discplineFilter	string	Discipline and subdiscipline filters See Discipline- and Objecttype Filters [→ 35]
	objecttypeFilter	string	Type and subtype filters See Discipline- and Objecttype Filters [→ 35]
	aliasFilter	string	Case-sensitive alias filter
Response	HTTP 200 OK A Page<Nodes, BrowserObject>-object (see Page<Name, Type [→ 50]>)		

Examples	**Request::**
	`GET /api/systembrowser/1?searchstring=QW5hbG9nKg&searchOption=1&caseSensitive=false&groupByParent=false`
	Response:
	<pre>{
	"Total": 1,
	"Page": 1,
	"Size": 200,
	"Nodes": [
	{
	"HasChild": false,
	"SystemId": 1,
	"ViewId": 9,</pre>		

```
      "Name": "1/1-AI_1",
      "Descriptor": "Analog Input 1",
      "Designation":
"System1.ManagementView:ManagementView.FieldNetworks.BAC0.
Hardware.1/1-3457.1/0-Local_IO.1/1-AI_1",
      "ObjectId": "System1:GmsDevice_1_1_1",
      "Attributes": {
        "DefaultProperty": "Present_Value",
        "ObjectId": "GmsDevice_1_1_1",
        "DisciplineDescriptor": "Building Automation",
        "DisciplineId": 50,
        "SubDisciplineDescriptor": "Unassigned",
        "SubDisciplineId": 0,
        "TypeDescriptor": "Sensor",
        "TypeId": 6700,
        "SubTypeDescriptor": "Unassigned",
        "SubTypeId": 0,
        "ManagedType": 80,
        "FunctionName": "Sensor"
      },
      "Location": "System1.Management View:Project.Field
Networks.BAC0.Hardware.Simulator Device 1.Local_IO.Analog
Input 1",
      "_links": []
    }
  ],
  "_links": [
    {
      "Rel": "page",
      "Href":
"api/systembrowser/1?searchString=QW5hbG9nKg&searchOption=
1&caseSensitive=False&groupByParent=False&pageSize=200&cur
rentPage={page}&disciplineFilter=null&objectTypeFilter=nul
l&aliasFilter=",
      "IsTemplated": true
    }
  ]
}
```


6.6.3.1 Searching with Wildcards

Two wildcard characters are supported in System Browser—the asterisk (*) and the question mark (?). Each functions differently. The asterisk wildcard serves as a placeholder for zero or more characters. The question mark wildcard serves as a placeholder for exactly one character only. Therefore, each wildcard serves different purposes.

: Allows you to add zero or more characters to your search criteria. For example, "a" matches and displays, “a”, "ab", "abc", and "abcd".

?: Allows you to add one character to your search. For example, "ab?" matches and returns "abc" but does not match or return "ab" and "abcd".

6.6.3.2 SearchOption

searchOption must have one of the following values.

searchOption	Description
0	Search in full name (Designation); default
1	Search in full description (Location)
2	Search in ObjectId
3	Search in Alias

6.6.3.3 Discipline and Objecttype Filters

This JSON string has two level filtering options–, group level (e.g. disciplines/types) and subgroup level (e.g. subdisciplines/subtypes). In case if SubGroupID/SubObjectTypeID is not provided along with GroupID/ObjectTypeID respectively then all the SubGroups/SubObjectTypes from the metioned Group/ObjectType will be taken for filtering.

- *Example*

```
{"<GroupID>":[<SubGroupID>,..,..,<SubGroupID>],"<GroupID>":[<SubGroupID>,..,..,<SubGroupID>]}
```

- *Concrete example*

```
{"0":[0,1,2,4],"20":[0,21,22]}
```

6.6.4 Search multiple Object Ids

Along with normal searches available, This API also provides bulk search option for searching multiple Object Ids.

Resource Path	/systembrowser/:systemId/:viewId		
HTTP Method	POST		
Parameters	systemId	uint	ID of a system **Required**
	viewId	uint	ID of the view in which search needs to be performed
	groupByParent	bool	If true, group search results by parent node
Body	Array of ObjectId	String[]	Array of object ids to be searched
Response	HTTP 200 OK Array of Multiple Object Ids search Browser Objects see [→ 55] An array of searched browser objects		

Examples	**Request:**
	`GET api/systembrowser/1/9?searchOption=2&caseSensitive=false&groupByParent=false`
	
	Post Data:
	`[`
	`  "System1:ManagementView_System_Settings",`
	`  "System1:JournalingRootFolder"`
	`]`
	
	Response:
	`[`
	`  {`		

```
    "ErrorCode": 0,
    "ObjectId": "System1:ManagementView_System_Settings",
    "Nodes": [
      {
        "HasChild": false,
        "SystemId": 1,
        "ViewId": 9,
        "Name": "SystemSettings",
        "Descriptor": "System Settings",
        "Designation":
"System1.ManagementView:ManagementView.SystemSettings",
        "ObjectId":
"System1:ManagementView_System_Settings",
        "Attributes": {
          "DefaultProperty":
"StatusPropagation.AggregatedSummaryStatus",
          "ObjectId": "ManagementView_System_Settings",
          "DisciplineDescriptor": "Management System",
          "DisciplineId": 0,
          "SubDisciplineDescriptor": "System Settings",
          "SubDisciplineId": 4,
          "TypeDescriptor": "View Element",
          "TypeId": 8000,
          "SubTypeDescriptor": "System Folder",
          "SubTypeId": 8014,
          "ManagedType": 153
        },
        "Location": "System1.Management
View:Project.System Settings",
        "_links": []
      }
    ]
  },
  {
    "ErrorCode": 0,
    "ObjectId": "System1:JournalingRootFolder",
    "Nodes": [
      {
        "HasChild": false,
        "SystemId": 1,
        "ViewId": 9,
        "Name": "JournalingRootFolder",
        "Descriptor": "Journaling",
        "Designation":
"System1.ManagementView:ManagementView.SystemSettings.Jour
nalingRootFolder",
        "ObjectId": "System1:JournalingRootFolder",
        "Attributes": {
          "DefaultProperty":
"StatusPropagation.AggregatedSummaryStatus",
          "ObjectId": "JournalingRootFolder",
          "DisciplineDescriptor": "Management System",
          "DisciplineId": 0,
          "SubDisciplineDescriptor": "System Settings",
          "SubDisciplineId": 4,
          "TypeDescriptor": "Journaling",
          "TypeId": 4500,
          "SubTypeDescriptor": "Journaling Printer",
          "SubTypeId": 4501,
```

```
          "ManagedType": 63
        },
        "Location": "System1.Management
View:Project.System Settings.Journaling",
        "_links": []
      }
    ]
  }
]
```


6.7 Value Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/values	x	
/valuessubscriptions	x	x

6.7.1 Retrieving a Value of an Object or Property

If no property is specified but only an object, the value of the default property is returned.

Resource Path	/values/:objectOrPropertyId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	objectOrPropertyId	string	Address of object or property **Required**
Response	HTTP 200 OK An array of ValueDetails objects [→62]		
Examples	**Request:** `GET /NBWSITest/NorthboundWebApiClient/api/values/U3lzdGVtMTpHbXNEZXZpY2VfMV8yOTEyOF8xLkRlc2NyaXB0aW9u` **Response:**		


```
[
  {
    "DataType": "BasicString",
    "Value": {
      "Value": "\"Analog Input 1\"",
      "Quality": "9439544818976425217",
      "QualityGood": true,
      "Timestamp": "2015-05-07T10:28:45.462Z"
    },
    "OriginalObjectOrPropertyId":
"System1:GmsDevice_1_29128_1.Description",
    "ObjectId": "GmsDevice_1_29128_1",
    "PropertyName": "Description",
    "AttributeId":
"System1:GmsDevice_1_29128_1.Description:_online.._value",
    "ErrorCode": 0,
    "IsArray": false,
    "_links": []
  }
]
```

6.7.2 Retrievig Values for a List of Objects or Properties

Resource Path	/values		
HTTP Method	POST		
Parameters			
Body	Array of ObjectOrPropertyId		
Response	HTTP 200 OK	An array of ValueDetails objects [→62]	

Examples	**Request:**
	`POST /NBWSITest/NorthboundWebApiClient/api/values`
	Post Data:
	`[ "GmsDevice_1_29128_16777217.Priority_Array", "System1:GmsDevice_1_29128_1.Description" ]`
	Response:
	`[{ "DataType": "BasicUint", "Value": { "Value": "4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,4294967295,0", "Quality": "9439544818973279489", "QualityGood": true, "Timestamp": "2015-05-07T10:36:08.834Z" }, "OriginalObjectOrPropertyId": "System1:GmsDevice_1_29128_16777217.Priority_Array", "ObjectId": "GmsDevice_1_29128_16777217", "PropertyName": "Priority_Array", "AttributeId": "System1:GmsDevice_1_29128_16777217.Priority_Array:_online.._value", "ErrorCode": 0, "IsArray": true, "_links": [] }, { "DataType": "BasicString", "Value": { "Value": "\"Analog Input 1\"", "Quality": "9439544818975376641", "QualityGood": true, "Timestamp": "2015-05-07T10:36:08.834Z" }, "OriginalObjectOrPropertyId": "System1:GmsDevice_1_29128_1.Description", "ObjectId": "GmsDevice_1_29128_1", "PropertyName": "Description", "AttributeId": "System1:GmsDevice_1_29128_1.Description:_online.._value", "ErrorCode": 0, "IsArray": false, "_links": [] }]`		

6.7.3 Creating a Subscription for a Change of a Value

Resource Path	/sr/valuessubscriptions/:connectionId		
HTTP Method	POST		
Parameters	**Property**	**Type**	**Description**
	connectionId	string	Identifier of subscription **Required**
	detailsRequired	bool	
Body	Array of ObjectOrPropertyId		
Response	HTTP 200 OK An array of Subscription objects [→56]		

Examples	**Request:**
	`POST /NBWSITest/NorthboundWebApiClient/api/sr/valuessubscriptions/977a6389-980c-41ca-84ab-2518fd0ca41a`
	
	Post Data:
	`[ "System1:GmsDevice_1_29128_1.Description" ]`
	Response:
	`[ { "Key": 1, "PropertyId": "System1:GmsDevice_1_29128_1.Description", "ErrorCode": 0, "_links": [ { "Rel": "unsubscribe", "Href": "api/sr/valuessubscriptions/977a6389-980c-41ca-84ab-2518fd0ca41a?subscriptionKey=%5b1%5d", "IsTemplated": false } ] } ]`		

6.7.3.1 Push Notification

After subscribing for changes of a value, the current value(s) will be sent in an initial notification. Afterwards, any change will be notified.

Type	**Name (case sensitive)**		
Hub	valuesHub		
Function	notifyValues(data)		
	Property	**Type**	**Description**
	data	Array of ValueDetails	See ValueDetails [→62]

6.7.4 Deleting a Subscription for a Change of a Value

Resource Path	/sr/valuessubscriptions/:connectionId		
HTTP Method	DELETE		
Parameters	**Property**	**Type**	**Description**
	connectionId	String	Identifier of subscription **Required**
	subscriptionkey	Array of int	List of subscription-keys
Response	HTTP 200 OK An array of unsubscription objects. See Link [→57]		

Examples	**Request:**
	`DELETE /NBWSITest/NorthboundWebApiClient/api/sr/valuessubscriptions/9d4211db-a740-44c3-93b8-015df1a529ca?subscriptionKey=[2,3]`
	Response:
	`[ { "ErrorCode": 0, "Key": 2 }, { "ErrorCode": 0, "Key": 3 } ]`		

6.8 Property Value Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/propertyvalues	x	

6.8.1 Retrieving Detailed Values for Object or Property Id

Retrieves properties with runtime values for an Object OR Property Id. This gives properties and attributes associated with the requested Object OR Property Id. If input is actually an Object Id then default property for the object id along with the attributes is returned in the response. If input is a property id then that specific property and attributes of that object id will be returned in the response. One optional Boolean type query string parameter readAllProperties is available in both the cases. If readAllProperties is set to True then all the properties along with the attributes are returned in the response for the requested Object OR Property Id

<table>
<tr><td>Resource Path</td><td colspan="3">/propertyvalues/:objectOrPropertyId</td></tr>
<tr><td>HTTP Method</td><td colspan="3">GET</td></tr>
<tr><td rowspan="3">Parameters</td><td>Property</td><td>Type</td><td>Description</td></tr>
<tr><td>objectOrPropertyId</td><td>string</td><td>ObjectOrPropertyId
Required</td></tr>
<tr><td>readAllProperties</td><td>bool</td><td>If True then all the properties for the requested Object OR Property Id will be returned in the response. Default is False.</td></tr>
<tr><td>Response</td><td colspan="3">HTTP 200 OK
The Object<Type> see [→ 62]</td></tr>
<tr><td>Examples</td><td colspan="3">Request:
GET api/propertyvalues/System1:HDB_ArchiveGroup_1
Response:
<pre>{
  "ErrorCode": 0,
  "ObjectId": "System1:HDB_ArchiveGroup_1",
  "Attributes": {
    "DefaultProperty":
"StatusPropagation.AggregatedSummaryStatus",
    "ObjectId": "HDB_ArchiveGroup_1",
    "DisciplineDescriptor": "Management System",
    "DisciplineId": 0,
    "SubDisciplineDescriptor": "Unassigned",
    "SubDisciplineId": 0,
    "TypeDescriptor": "Server Software",
    "TypeId": 6900,
    "SubTypeDescriptor": "History Database",
    "SubTypeId": 6903,
    "ManagedType": 0
  },
  "Properties": [
    {
      "Order": 0,
      "PropertyName":
"StatusPropagation.AggregatedSummaryStatus",
      "Descriptor": "Summary Status",
      "Type": "ExtendedEnum",
      "Usage": 7,
      "Value": {
        "Value": "0",
        "DisplayValue": "Normal",
        "Quality": "9439544818969084161",
        "QualityGood": true,
        "Timestamp": "2015-05-21T05:08:56.772Z"
      },
      "Resolution": 0,
      "PropertyAbsent": false,
      "IsArray": false
    }
  ]
}</pre></td></tr>
</table>

6.9 Properties Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/properties	x	

6.9.1 Retrieving Detailed Values for Object or Property Id

Retrieves properties without any runtime values for an Object OR Property Id. If input is actually an Object Id then default property for the object id will be returned in the response. If input is a property id then that specific property will be returned in the response. Two optional query string parameters are available in this interface. First is requestType of type int, which specifies whether the request is to read Property Names, Properties, Attributes, Properties and Attributes. Second parameter is readAllProperties of type bool, if this isTrue then all the properties for the requested Object OR Property Id will be returned in the response.

Resource Path	/properties/:objectOrPropertyId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	objectOrPropertyId	string	ObjectOrPropertyId **Required**
	requestType	Int	Specifies the request type. See [→ 44]. Default is 2.
	readAllProperties	bool	If True then all the properties for the requested Object OR Property Id will be returned in the response. Default is False.
Response	HTTP 200 OK The Object<Type> see [→ 62]		
	requestType	**Return Type**	**Description**
	0	See [→ 67]	Property collections contain list of all property-names
	1	Object<null>	Property collections empty; object contains attributes
	2 (default)	Object<PropertyDetails>	Property collections populated; no attribute field
	3	Object<PropertyDetails>	Property collections populated; object contains attributes

Examples	**Request1: Get Property Names**
	`GET api/properties/System1:HDB_ArchiveGroup_1?requestType=0`
	Response1:
	`{`
	`  "ObjectId": "System1:HDB_ArchiveGroup_1",`
	`  "Properties": [`
	`  "Export.Activated",`
	`  "Max_Retention.Activated",`
	`  "Max_Retention.Delay_Span",`
	`  "Max_Retention.Delay_Unit",`
	`  "Max_Retention.FirstDate",`		

```
    "Max_Retention.Invocation_Period",
    "Max_Retention.Invocation_Unit",
    "Max_Retention.Max_Count",
    "Min_Retention.Delay_Span",
    "Min_Retention.Delay_Unit",
    "Min_Retention.Min_Count",
    "StatusPropagation.AggregatedSummaryStatus"
  ],
  "FunctionProperties": []
}
```

Request2: Get Properties and Attributes

```
GET api/properties/System1:HDB_ArchiveGroup_1?requestType=3
```

Response2:

```
[
  {
    "ErrorCode": 0,
    "ObjectId": "System1:HDB_ArchiveGroup_1",
    "Attributes": {
      "DefaultProperty":
"StatusPropagation.AggregatedSummaryStatus",
      "ObjectId": "HDB_ArchiveGroup_1",
      "DisciplineDescriptor": "Management System",
      "DisciplineId": 0,
      "SubDisciplineDescriptor": "Unassigned",
      "SubDisciplineId": 0,
      "TypeDescriptor": "Server Software",
      "TypeId": 6900,
      "SubTypeDescriptor": "History Database",
      "SubTypeId": 6903,
      "ManagedType": 0
    },
    "Properties": [
      {
        "Order": 0,
        "PropertyName":
"StatusPropagation.AggregatedSummaryStatus",
        "Descriptor": "Summary Status",
        "Type": "ExtendedEnum",
        "Usage": 7,
        "Resolution": 0,
        "PropertyAbsent": false,
        "IsArray": false
      }
    ]
  }
]
```

6.9.1.1 Detail Request Type

requestType must have one of the following values.

requestType	Description
0	Return property names only.
1	Return attributes of object only.
2	Return properties with detailed information; default.
3	Return properties with detailed information and attributes of object.

6.9.2 Retrieving Detailed Values for multiple Object or Property Ids bulk interface

This interface is same as that of 7.9.1 only thing is this provides bulk interface to read properties and attributes of multiple Object OR Property Ids in the same request using HTTP POST. Array of Object OR Property Ids is passed in the POST data. Rest all the configuration is same as that of 7.9.1.

<table>
<tr><td>Resource Path</td><td colspan="3">/ properties</td></tr>
<tr><td>HTTP Method</td><td colspan="3">POST</td></tr>
<tr><td rowspan="3">Parameters</td><td>Property</td><td>Type</td><td>Description</td></tr>
<tr><td>requestType</td><td>Int</td><td>Specifies the request type. See [➞ 44]. Default is 2.</td></tr>
<tr><td>readAllProperties</td><td>bool</td><td>If True then all the properties for the requested Object OR Property Id will be returned in the response. Default is False.</td></tr>
<tr><td>Body</td><td colspan="3">Array of Object OR Property Ids</td></tr>
<tr><td rowspan="4">Response</td><td colspan="3">HTTP 200 OK
The Object<Type> see [➞ 62]</td></tr>
<tr><td>requestType</td><td>Return Type</td><td>Description</td></tr>
<tr><td>1</td><td>Object<null></td><td>Property collections empty; object contains attributes</td></tr>
<tr><td>2</td><td>Object<PropertyDetails></td><td>Property collections populated; no attribute field</td></tr>
<tr><td>Examples</td><td colspan="3">Request:
<code>Post api/properties?requestType=3</code>

Request Body:
<code>["System1:HDB_ArchiveGroup_1.Min_Retention.Delay_Span","System1:HDB_ArchiveGroup_1.Min_Retention.Delay_Unit"]</code>
Reaponse:
<code>[
{
"ErrorCode": 0,
"ObjectId": "System1:HDB_ArchiveGroup_1",
"Attributes": {
"DefaultProperty":
"StatusPropagation.AggregatedSummaryStatus",</code></td></tr>
</table>

```
      "ObjectId": "HDB_ArchiveGroup_1",
      "DisciplineDescriptor": "Management System",
      "DisciplineId": 0,
      "SubDisciplineDescriptor": "Unassigned",
      "SubDisciplineId": 0,
      "TypeDescriptor": "Server Software",
      "TypeId": 6900,
      "SubTypeDescriptor": "History Database",
      "SubTypeId": 6903,
      "ManagedType": 0
    },
    "Properties": [
      {
        "Order": 3,
        "PropertyName": "Min_Retention.Delay_Span",
        "Descriptor": "Min Retention Span",
        "Type": "ExtendedInt",
        "Usage": 11,
        "Min": 1,
        "Resolution": 0,
        "PropertyAbsent": false,
        "IsArray": false
      }
    ]
  },
  {
    "ErrorCode": 0,
    "ObjectId": "System1:HDB_ArchiveGroup_1",
    "Attributes": {
      "DefaultProperty":
"StatusPropagation.AggregatedSummaryStatus",
      "ObjectId": "HDB_ArchiveGroup_1",
      "DisciplineDescriptor": "Management System",
      "DisciplineId": 0,
      "SubDisciplineDescriptor": "Unassigned",
      "SubDisciplineId": 0,
      "TypeDescriptor": "Server Software",
      "TypeId": 6900,
      "SubTypeDescriptor": "History Database",
      "SubTypeId": 6903,
      "ManagedType": 0
    },
    "Properties": [
      {
        "Order": 4,
        "PropertyName": "Min_Retention.Delay_Unit",
        "Descriptor": "Min Retention Unit",
        "Type": "ExtendedEnum",
        "Usage": 11,
        "Max": 7,
        "Resolution": 0,
        "PropertyAbsent": false,
        "IsArray": false
      }
    ]
  }
]
```

6.10 Command Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/commands	x	
/commandssubscriptions	x	x

6.10.1 Retrieving a List of Commands for a Provided Property

Some commands might be non executable at this moment.

Resource Path	/commands/:propertyId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	propertyId	string	PropertyId **Required**
	commandId	string	Optional filter for a specific commandId
	enabledCommandsOnly	bool	If set to true only currently enabled commands will be returned (default false).
	clientType	string	If set commands can be filtered for specific clients (All, Headless, Headful)
Response	HTTP 200 OK An array of PropertyCommand objects See [→ 52]		

Examples	**Request:**
	`GET api/commands/U3lzdGVtMTpSZXBvcnREZWZhdWx0Rm9sZGVyLk1heFJldGVudGlvblVuaXQ`
	Response:
	`{ "PropertyId": "System1:GmsDevice_1_27739_1@[]Value", "ErrorCode": 0, "Commands": [{ "PropertyId": "System1:GmsDevice_1_27739_1@[]Value", "Descriptor": "Command", "Parameters": [{ "Name": "Value", "DataType": "ExtendedReal", "DefaultValue": "0", "Order": 0, "Descriptor": "Value", "Max": 999, "Min": -999, "EnumerationTexts": [] }], "Id": "Write", "_links": [{ "Rel": "self", "Href": "api/commands/System1%3aGmsDevice_1_27739_1%40%5b%5dValue/`

	<pre>Write",
	"IsTemplated": false
	}
	]
	}
	]
	},</pre>

6.10.2 Retrieving Lists of Commands for a List of Properties

Resource Path	/commands		
HTTP Method	POST		
Parameters	**Property**	**Type**	**Description**
	propertyId	string	PropertyId **Required**
	commandId	string	Optional filter for a specific commandId
	enabledCommandsOnly	bool	If set to true only currently enabled commands will be returned (default false).
	clientType	string	If set commands can be filtered for specific clients (All, Headless, Headful)
Body	Array of PropertyId		
Response	HTTP 200 OK An aArray of PropertyCommand objects See [→ 52]		

Examples	**Request:**
	Post : /api/commands/
	Reaponse:
	<pre>List of Commands fetched for two different PropertyId:
	[
	{
	"PropertyId": "System1:GmsDevice_1_27739_1@[]Value",
	"ErrorCode": 0,
	"Commands": [
	{
	"PropertyId":
	"System1:GmsDevice_1_27739_1@[]Value",
	"Descriptor": "Command",
	"Parameters": [
	{
	"Name": "Value",
	"DataType": "ExtendedReal",
	"DefaultValue": "0",
	"Order": 0,
	"Descriptor": "Value",
	"Max": 999,
	"Min": -999,
	"EnumerationTexts": []
	}
	],
	"Id": "Write",
	"_links": [
	{
	"Rel": "self",</pre>		

```
            "Href":
"api/commands/System1%3aGmsDevice_1_27739_1%40%5b%5dValue/
Write",
            "IsTemplated": false
          }
        ]
      }
    ]
  },
  {
    "PropertyId":
"System1:ManagementView_ManagementSystem_Servers_Server.Ba
ckup.Status",
    "ErrorCode": 0,
    "Commands": [
      {
        "PropertyId":
"System1:ManagementView_ManagementSystem_Servers_Server.Ba
ckup.Status",
        "Descriptor": "Start",
        "Parameters": [],
        "Id": "Start",
        "_links": [
          {
            "Rel": "self",
            "Href":
"api/commands/System1%3aManagementView_ManagementSystem_Se
rvers_Server.Backup.Status/Start",
            "IsTemplated": false
          }
        ]
      },
      {
        "PropertyId":
"System1:ManagementView_ManagementSystem_Servers_Server.Ba
ckup.Status",
        "Descriptor": "Cancel",
        "Parameters": [],
        "Id": "Cancel",
        "_links": [
          {
            "Rel": "self",
            "Href":
"api/commands/System1%3aManagementView_ManagementSystem_Se
rvers_Server.Backup.Status/Cancel",
            "IsTemplated": false
          }
        ]
      }
    ]
  },
  {
    "PropertyId":
"System1:GmsDevice_1_1_1542.Present_Value",
    "ErrorCode": 0,
    "Commands": []
  }
]
```

6.10.3 Executing a Command

Resource Path	/commands/:propertyId/:commandId		
HTTP Method	POST		
Parameters	**Property**	**Type**	**Description**
	propertyId	string	PropertyId **Required**
	commandId	string	Command identifier **Required**
	commandInput	CommandInputForExecution	command parameter details (Name, Value pair) sent from client if required.
Body	An array of NameValue objects of Type(CommandInputForExecution)		
Response	HTTP 200 OK		

Examples	**Request:**
	`POST api/commands/U3lzdGVtMTpSZXBvcnREZWZhdWx0Rm9sZGVyLk1heFJldGVudGlvblVuaXQ/Write`
	Response:
	`HTTP 200 OK`		

6.10.4 Creating a Subscription for a Change of a Command

Resource Path	/sr/commandssubscriptions/:connectionId		
HTTP Method	POST		
Parameters	**Property**	**Type**	**Description**
	connectionId	string	Identifier of subscription **Required**
	propertyIdList	String[]	Array of Properties from POST data, which will be subscribed for Command Change notification
Body	Array of PropertyId		
Response	HTTP 200 OK An array of Subscription objects See Link [→ 46]		

Examples	**Request:**
	`Post :api/sr/commandssubscriptions/87a73827-f25b-41e8-9225-2f08316ea0fb`
	Response:
	`{ "Key": 1, "PropertyId": "System1:ManagementView_ManagementSystem_Servers_Server.Backup.Status", "ErrorCode": 0, "_links": [ { "Rel": "unsubscribe", "Href": "api/sr/commandssubscriptions/87a73827-f25b-41e8-9225-2f08316ea0fb?subscriptionKey=%5b1%5d", "IsTemplated": false } ] } ]`		

6.10.4.1 Push Notification

After subscribing for changes of a command(s) **all** command(s) will be sent in an initial notification. Afterwards any change will be notified.

Type	**Name (case sensitive)**		
Hub	commandsHub		
Function	notifyCommands(data)		
	Property	**Type**	**Description**
	data	Array of PropertyCommand	See PropertyCommand [→ 66]

6.10.5 Deleting a Subscription for a Change of a Command.

Resource Path	/sr/commandssubscriptions/:connectionId		
HTTP Method	DELETE		
Parameters	**Property**	**Type**	**Description**
	connectionId	String	Identifier of subscription **Required**
	subscriptionkey	Array of int	List of subscription keys
Response	HTTP 200 OK An array of unsubscription objects. See Link [→56]		

Examples	**Request:** Delete
	`api/sr/commandssubscriptions/a43e4363-296e-405d-977b-36dec9ddbafc?subscriptionKey=[1,2]`
	Response:
	`{ "ErrorCode": 0, "Key": 1 }, { "ErrorCode": 0, "Key": 2 }`		

6.11 Trend Service

Resource	**Requires Authentication**	**Requires SignalR (Push Notifications)**
/trendseriesinfo	x	
/trendseries	x	

6.11.1 Retrieving a List of All Trend Collector Objects

Retrieve a list of all Trend Collector objects of a given system.

Resource Path	/trendseriesinfo/:systemId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	systemId	uint	ID of a system **Required**

Response	HTTP 200 OK An array of TrendCollector objects [→64]

Examples	**Request:**
	`GET /api/trendseriesinfo/1`
	Response:
	`[{ "ObjectId": "System1:GmsDevice_1_29129_4194305", "PropertyName": "Present_Value", "CollectorObjectOrPropertyId": "System1:GmsDevice_1_29129_113246209", "TrendseriesId": "System1:GmsDevice_1_29129_113246209.general.Data:_original.2._value" }, { "ObjectId": "System1:GmsDevice_1_29129_4194305", "PropertyName": "Present_Value", "CollectorObjectOrPropertyId": "System1:GmsDevice_1_29129_113246210", "TrendseriesId": "System1:GmsDevice_1_29129_113246210.general.Data:_original.2._value" }]`

6.11.2 Retrieving a List of Trend Collector Objects

Retrieve a list of Trend Collector objects. An object ID of a trend object (e.g. TrendLog/TrendLogMultiple) or a property ID of a trended object is to be provided.

Resource Path	/trendseriesinfo/:objectOrPropertyId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	objectOrPropertyId	string	ObjectOrPropertyId **Required**
Response	HTTP 200 OK An array of TrendCollector objects [→64]		

Examples	**Request:**
	`GET api/trendseriesinfo/U3lzdGVtMTpHbXNEZXZpY2VfMV8yOTEyOV80MTk0MzA1`
	Response:
	`[{ "ObjectId": "System1:GmsDevice_1_29129_4194305", "PropertyName": "Present_Value", "CollectorObjectOrPropertyId": "System1:GmsDevice_1_29129_113246209", "TrendseriesId": "System1:GmsDevice_1_29129_113246209.general.Data:_original.2._value" }, { "ObjectId": "System1:GmsDevice_1_29129_4194305", "PropertyName": "Present_Value",`		

	<pre> "CollectorObjectOrPropertyId":
	"System1:GmsDevice_1_29129_113246210",
	    "TrendseriesId":
	"System1:GmsDevice_1_29129_113246210.general.Data:_original.2._value"
	  }
	]</pre>

6.11.3 Retrieving Borders of a Trend Series

Resource Path	/trendseries/:trendseriesId/borders		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	trendseriesId	string	Trend series identifier **Required**
Response	HTTP 200 OK A TrendBorder object [→64]		

Examples	**Request:**
	<pre>GET
	/NBWSITest/NorthboundWebApiClient/api/trendseries/U3lzdGVtMTpHbXNEZXZpY2VfMV8yOTEyOV84Mzg4NjA4MS5nZW5lcmFsLkRhdGE6X29mZmxpbmUuLl92YWx1ZQ/borders</pre>**Response:**
	<pre>{
	  "From": "2015-05-13T12:00:34.37Z",
	  "To": "2015-05-13T12:06:57.9Z"
	}</pre>		

6.11.4 Retrieving a List of Trends

Resource Path	/trendseries/:trendseriesId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	trendseriesId	string	Trend series identifier **Required**
	from	DateTime	Start time of trend series **Required**
	to	DateTime	End time of trend series **Required**
	intervals	ushort	If provided, the result contains the min. and max. value per interval.
	addBonusValue	bool	If true, adds one additional value on the left side of the from-border and one on the right side of the to-border. Default: true
	addDescriptor	bool	If true, adds localized value (if applicable) to result. Default: false
Response	HTTP 200 OK A TrendSeries object [→64]		
Examples	**Request:** <pre>GET</pre>		

	/NBWSITest/NorthboundWebApiClient/api/trendseries/U3lzdGVtMTpHbXNEZXZpY2VfMV8yOTEyOV84Mzg4NjA4MS5nZW5lcmFsLkRhdGE6X29mZmxpbmUuLl92YWx1ZQ?from=2015-05-13T12:05:53.000Z&to=2015-05-13T12:05:54.000Z&intervals=5&addBonusValue=true&description=true
	Response:
	{
	"Id": "System1:GmsDevice_1_29129_83886081.general.Data:_offline._value",
	"SeriesPropertyId": "System1:GmsDevice_1_29129_16777217.Present_Value",
	"Series": [
	{
	"Value": "1",
	"DisplayValue": "ACTIVE",
	"Quality": "8589934592",
	"QualityGood": true,
	"Timestamp": "2015-05-13T12:05:52.59Z"
	},
	{
	"Value": "1",
	"DisplayValue": "INACTIVE",
	"Quality": "8589934592",
	"QualityGood": true,
	"Timestamp": "2015-05-13T12:05:55.59Z"
	}
	]
	}

6.12 Diagnostics Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/diagnostics		

6.12.1 Test Reachability of the Web Service Interface

A test whether or not the Web Service Interface replies to requests.

Resource Path	/diagnostics
HTTP Method	GET
Response	HTTP 200 OK
Examples	**Request:** GET /NBWSITest/NorthboundWebApiClient/api/diagnostics/ **Response:** HTTP 200 OK

6.13 Language Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/languages	x	

6.13.1 Retrieving the Language of the Logged in User

Resource Path	/languages
HTTP Method	GET
Response	HTTP 200 OK A Language object [→64]

Examples	**Request:**
	`GET /api/Languages?api_key=username:password`
	Response:
	`{`
	`  "Descriptor": "English (United States)",`
	`  "Code": "en-US"`
	`}`

6.14 Image Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/images	x	

6.14.1 Retrieving an Image

NOTE:
In order to successfully process this request, the image needs to be present in a feasible format and size in the system.

Resource Path	/images/:imageId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	imageId	string	Image name (file name without extension) **Required**
	path	string	Path within a library **Required**
	format	string	Image format (png or svg) **Required**
	width	int	Image width in pixels
	height	int	Image height in pixels
	encodeAsBase64	bool	If true, the image is returned as a base64-encoded string.
Response	HTTP 200 OK The returned data depends on encodeAsBase64.		
	encodeAsBase64	**Return Type**	**Description**
	false (default)	image (image media type; see [10] [➔ 12])	Image in the requested format
	true	string	base64-encoded string
Examples	**Request:** `/api/images/Evt_Discp_Infrastr_None_001?format=png&width=1&height=1&path=libraries%5CGlobal_Base_HQ_1%5Cicons&encodeAsBase64=true`		

	Response: `iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAYAAAAfFcSJAAAAAXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAADsIAAA7CARUoSoAAAAANSURBVBhXY/j//78GAAkkAyb6sRXrAAAAAElFTkSuQmCC`

6.15 Tables Service

Resource	Requires Authentication	Requires SignalR (Push Notifications)
/tables	x	

6.15.1 Retrieving a Table or an Entry of a Table

Resource Path	/tables/:tableId/:tableEntryId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	tableId	string	Table identifier (table name) **Required**
	id	string	Table entry identifier (key of row of table)
Response	HTTP 200 OK The returned data depends whether or not id is provided.		
	id	**Return Type**	**Description**
	Not provided	Array of KeyValue<*keyname*,string>	List of all key-value pairs in this table
	Provided	string	Localized value of this column

Examples	**Request1: id Not provided**
	`GET api/tables/disciplines`
	Response1:
	`{ "0": "Management System", "20": "Building Infrastructure", "50": "Building Automation", "100": "Fire", "150": "Security", "200": "Energy Management", "250": "Notification" }`
	
	Request2: id provided
	`GET api/tables/disciplines/100`
	Response2:
	`"Fire"`		

6.15.2 Retrieving a Subtable of a Table

Resource Path	/tables/:tableId/subtables/:subtableId		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	tableId	string	Table identifier (table name) **Required**

	subtableId	string	Subtable identifier (table name) **Required**
Response	HTTP 200 OK A list of all key-value pairs in this table as array of KeyValue<*keyname*,string>		

Examples	**Request:**
	`GET api/tables/disciplines/subtables/icons`
	Response:
	`{ "0": "libraries\\Global_Base_HQ_1\\icons\\Evt_Discp_System_None_001.xaml|Evt_Discp_System_None_001", "20": "libraries\\Global_Base_HQ_1\\icons\\Evt_Discp_Infrastr_None_001.xaml|Evt_Discp_Infrastr_None_001", "50": "libraries\\Global_Base_HQ_1\\icons\\Evt_Discp_BA_None_001.xaml|Evt_Discp_BA_None_001" }`		

6.15.3 Retrieving Subgroups with filter

This method help retrieves subgroups of the Disciplines OR ObjectTypes. Filter is also available for retrieving specific Disciplines, SubDisciplines OR ObjectTypes, SubObjectTypes. This filter parameter is query string parameter. If no SubDisciplines/SubObjectTypes are specified while requesting Disciplines/ObjectTypes respectively then all the SubDisciplines/SubObjectTypes which comes under the requested Discipline/ObjectType will be returned. Otherwise only the requested SubDisciplines/SubObjectTypes will be returned.

Resource Path	/tables/:tableId/subgroups		
HTTP Method	GET		
Parameters	**Property**	**Type**	**Description**
	tableId	string	Table identifier (table name) **Required**
	filter	string	This is query string parameter. Filter for Disciplines, SubDisciplines OR ObjectTypes, SUbObjectTypes **Required**
Response	HTTP 200 OK Based on the request is for Disciplines OR ObjectTypes response will be returned. 1. Disciplines : See [→ 63] 2. ObjectTypes : See [→ 63]		

Examples	**Request1: Disciplines and SubDisciplines request**
	`GET api/tables/disciplines/subgroups?filter={"0":[0,1],"100":[101,102]}`
	Response1:
	`[ { "DisciplineId": 0, "DisciplineDescriptor": "Management System", "SubDisciplines": [ { "Id": 0, "Descriptor": "Unassigned" },`		

```
      {
        "Id": 1,
        "Descriptor": "Applications"
      }
    ]
  },
  {
    "DisciplineId": 100,
    "DisciplineDescriptor": "Fire",
    "SubDisciplines": [
      {
        "Id": 101,
        "Descriptor": "Evacuation"
      },
      {
        "Id": 102,
        "Descriptor": "Extinguishing"
      }
    ]
  }
]
```


Request2: ObjectTypes and SubObjectTypes request

```
GET
api/tables/objecttypes/subgroups?filter={"1000":[1001,1002],"
1100":[1101,1102]}
```

Response2 :


```
[
  {
    "ObjectTypeId": 1000,
    "ObjectTypeDescriptor": "Client Software",
    "SubObjectTypes": [
      {
        "Id": 1001,
        "Descriptor": "Control Room"
      },
      {
        "Id": 1002,
        "Descriptor": "Mobile"
      }
    ]
  },
  {
    "ObjectTypeId": 1100,
    "ObjectTypeDescriptor": "Command",
    "SubObjectTypes": [
      {
        "Id": 1101,
        "Descriptor": "Analog"
      },
      {
        "Id": 1102,
        "Descriptor": "Binary"
      }
    ]
  }
]
```

7 Objects and Data Types

7.1 Common

7.1.1 Link

Models a hyperlink to another resource.

Property	Type	Description
Href	string	Hyperlink to a resource (URI [RFC3986] or URI template [RFC6570])
IsTemplated	bool	If true, Href contains a placeholder in its link.
Rel	string	Link relation type (RFC5988)

7.1.2 Page<Name, Type>

Property	Type	Description
_links	Array of Link	See Link [→ 50]
<Name>	Array of <Type>	Name and type depend on current instance
Page	int	Number of current page (first page = 1)
Size	int	Number of items per page
Total	Int	Total number of items

7.1.3 KeyValue<Name, Type>

Property	Type	Description
<Name>	<Type>	Name and type depend on current instance

7.1.4 NameValue

Property	Type	Description
DataType	ApiDataType	Data type of the property
Name	string	Name of the property
Value	string	Value of the property

7.1.5 Subscription

Property	Type	Description
_links	Array of Link	See Link [→ 58]
Key	int	Subscription key is specific for the requested ID
ErrorCode	int	If value is >0, an error occurred.
PropertyId	string	Address of a property

7.1.6 UnSubscription

Property	Type	Description
Key	int	Subscription key is specific for the requested ID
ErrorCode	int	If value is >0, an error occurred.

7.1.7 Value

Property	Type	Description
Value	string	Raw value
DisplayValue	string	Value for display purpose (textual representation of the raw value)
Timestamp	DateTime	Timestamp of this value
QualityGood	bool	
Quality	string	See Quality Flags [→ 59]

7.1.7.1 Quality Flags

Quality is a string representation of a 64-bit bit-set. The string contains a ulong numeric value. Bits not listed are not in use.

Bit	Description	QualityGood	Efffect on Value
0	variable active/inactive	TRUE	value ok
1	default value set explicitly	TRUE	value ok
2	default value set automatically	TRUE	value ok
3	WinCC OA value range violated	FALSE	value not ok
4	value of the variable "out of range"	FALSE	value not ok
5	explicit invalid	FALSE	value not ok
6	invalid set by an interface driver	FALSE	value not ok
8	default value invalid	TRUE	value ok
9	set during general query	TRUE	value ok
10	set during an individual query	TRUE	value ok
11	interface driver active	TRUE	value ok
12	value corrected	TRUE	value ok
13	value condensed or compressed, used (HDB & RAIMA) with archiving/compression.	TRUE	value ok
14	corrected value condensed	TRUE	value ok
15	additional correction value	TRUE	value ok
16	compressed value invalid	TRUE	value ok
17	source time invalid (corrected by the Event Manager)	TRUE	value ok
19	disable last value storage	TRUE	value ok
20	value changed	TRUE	value ok
21	value up (or set to same value)	TRUE	value ok
32	driver out of service	TRUE	value ok
33	driver alarm	TRUE	value ok
34	driver fault	TRUE	value ok
35	driver overridden	TRUE	value ok

36	driver subscribed	TRUE	value ok
37	driver property invalid	FALSE	value not ok
40	trend: time shift	FALSE	no value, is event
41	trend: log enabled	FALSE	no value, is event
42	trend: error	FALSE	no value, is event
43	trend: purge	FALSE	no value, is event
44	trend: rollover	FALSE	no value, is event
45	trend: value is status	FALSE	value is status/error
46	trend: log interrupted	FALSE	no value, is event
48	trend: start logging	TRUE	value ok
49	trend: value reduced	TRUE	value ok
50[1]	priority 1 (0 bit; 2^0)	TRUE	value ok
51[1]	priority 2 (1 bit; 2^1)	TRUE	value ok
52[1]	priority 4 (2 bit; 2^2)	TRUE	value ok

[1] Bit 50 / Bit 51 / Bit 52 are to be combined into a single value (priority 0..7)

7.1.8 Attributes

Property	Type	Description
DefaultProperty	string	Default property for object
DisciplineDescriptor	string	Localized name of discipline
DisciplineId	int	ID of discipline
FunctionName	string	
ManagedType	string	
ObjectId	string	
SubDisciplineDescriptor	string	Localized name of subdiscipline
SubDisciplineId	int	ID of subdiscipline
SubTypeDescriptor	string	Localized name of subtype
SubTypeId	int	ID of subtype
TypeDescriptor	string	Localized name of type
TypeId	int	ID of type

7.1.9 ApiDataType

Name	Description	BaseType
None		
BasicChar		char
BasicUint		uint
BasicInt		int
BasicFloat		double
BasicBool		boolean
BasicBit32		uint
BasicString		string
BasicTime		DateTime
BasicObjectOrPropertyId		string
BasicLangText		string

BasicBlob		string
ExtendedBool	Extends BasicBool with a language text for true/false.	
ExtendedInt	Extends BasicInt with Min, Max, Default, Unit.	
ExtendedUint	Extends BasicUint with Min, Max, Default, Unit.	
ExtendedReal	Extends BasicFloat with Min, Max, Default, Unit.	
ExtendedEnum	Extends BasicInt with enumeration texts.	
ExtendedBitString	Extends BasicInt with bit set texts.	
ExtendedDateTime	Extends BasicString with semantics for a DateTime string format.	
ExtendedApplSpecific	Extends BasicString with the knowledge that the string has a dedicated meaning for some application. Typically, the string will be XML-encoded.	
ExtendedAny	Extends BasicFloat with the permission to switch its type.	
ExtendedComplex	Extends BasicBlob.	
ExtendedDuration	Extends BasicUint with Min, Max, Default and time duration specific configurations.	

7.1.10 Subgroups

Property	Type	Description
Id	Uint	Id of the SubGroup
Descriptor	string	Descriptor of the SubGroup

7.2 Token Service

7.2.1 Login

Property	Type	Description
access_token	string	Token to be provided for requesting protected resources.
expires_in	int	Seconds until token expires at the latest **NOTE**: Token will expire sooner in case of inactivity; see Authentication [→ 15].
token_type	string	Only **Bearer** is supported at this time.

7.3 Event Service

7.3.1 Event

Property	Type	Description
_links	Array of Link	See Link [→ 58]
CategoryDescriptor	string	Localized textual representation of CategoryId
CategoryId	int	Alarm category
Cause	string	Cause came with alarm (localized)
Commands	Array of EventCommand	List of available commands. See Event Command [→ 63]
CreationTime	DateTime	Time when alarm created.

Deleted	bool	Flag whether an event was deleted. This is for push notifications only.
Direction	EventDirection	See EventDirection [→ 63]
EventId	uint	Event counter (non-unique)
Id	string	Desigo CC alarm identifier → Invariant across sessions
InfoDescriptor	string	Additional information to an event
SrcDescriptor	string	Tree node descriptor
SrcDesignation	string	Concatenated tree node names
SrcDisciplineDescriptor	string	Localized textual representation of DisciplineId
SrcDisciplineId	int	Alarm discipline
SrcLocation	string	Concatenated tree-node descriptors
SrcName	string	Tree node name
SrcObservedPropertyId	string	Observed property which lets SrcPropertyId trigger an event
SrcPropertyId	string	Source of an event
SrcState	SourceState	See SourceState [→ 62]
SrcSystemId	uint	Unique system identifier
SrcViewDescriptor	string	View descriptor
SrcViewName	string	View name
State	EventState	See EventState [→ 62]

7.3.1.1 SourceState

SrcState must have one of the following values.

SrcState	Description
Active	Object is active
Quiet	Object is quiet

7.3.1.2 EventState

State must have one of the following values.

State	Description
Unprocessed	Unprocessed
ReadyToBeReset	Ready to be reset
ReadyToBeClosed	Ready to be closed
WaitingOPCompletion	Waiting for completion
Acked	Acknowledged
WaitingForCommandExecution	Waiting for command execution
Closed	Closed

7.3.1.3 EventDirection

Direction must have one of the following values.

Direction	Description
In	IN alarm
Out	OUT alarm
None	Neither IN nor OUT alarm

7.3.2 EventCommand

Property	Type	Description
_links	Array of Link	See Link [→ 58]
EventId	string	Unique identifier of an event
Id	EventCommandId	See Event Command ID [→ 27]

7.4 Event Counter Service

7.4.1 EventCounterList

Property	Type	Description
EventCategoryCounters	Array of EventCounter	Categorized list of event counters
TotalCounters	int	Total counters in the system for logged-in user
TotalUnprocessedCounters	int	Total unprocessed counters in the system for logged-in user

7.4.2 EventCounter

Property	Type	Description
_links	Array of Link	See Link [→ 58]
CategoryId	int	Category key
CategoryDescriptor	string	Localized category name
TotalCount	int	Total events of this category
UnprocessedCount	int	Total unprocessed events of this category

7.5 System Browser Service

7.5.1 View

Property	Type	Description
_links	Array of Link	See Link [→ 50]
Designation	string	Full view name
Descriptor	string	Localized view name
Name	string	View name
SystemId	uint	Unique ID of a system
ViewId	uint	Unique ID of a view within a system

7.5.2 BrowserObject

Property	Type	Description
_links	Array of Link	See Link [→ 50]
Attributes	Attributes	See Attributes [→ 51]
Descriptor	string	Localized browser object name
Designation	string	Full view name
HasChild	bool	Flag whether child nodes are available
Name	string	View name
Location	string	Localized full browser object name
ObjectId	string	Node address
SystemId	uint	Unique ID of a system
ViewId	uint	Unique ID of a view within a system

7.5.3 Multiple ObjectId search BrowserObject

Property	Type	Description
ErrorCode	int	If value is >0, an error occurred
ObjectId	string	Searched Object Id
Nodes	Array of Browser Objects	Array of Browser Objects returned as search result for ObjectId see [→ 55]

7.6 Value Service

7.6.1 ValueDetails

Property	Type	Description
_links	Array of Link	See Link [→ 58]
AttributeId	string	Address of attribute to which value belongs to
DataType	ApiDataType	
ErrorCode	int	If value is >0, an error occurred

ObjectId	string	Address of object to which value belongs to
OriginalObjectOrPropertyId	string	Address of requested object or property
PropertyName	string	Name of property
Value	Value	See Value [→ 59]
IsArray	bool	Flag whether value is array or not

7.7 PropertyValue Service

7.7.1 Object<Type>

Property	Type	Description
_links	Array of Link	See Link [→ 58]
Attributes	Attributes	See Attributes [→ 60]
ErrorCode	int	If value is >0, an error occurred.
ObjectId	string	Address of object to which value belongs to
Properties	Array of <Type>	
FunctionProperties	Array of <Type>	

7.7.2 PropertyDetails

Property	Type	Description
_links	Array of Link	See Link [→ 58]
Descriptor	string	
Max	string	Allowed maximum value; empty if maximum value of datatype
Min	string	Allowed minimum value; empty if minimum value of data type
Order	int	Order of the properties
PropertyName	string	The property's name. Name is invariant and serves as property identifier.
Resolution	int	
Type	ApiDataType	
UnitDescriptor	string	Localized representation of unit
UnitId	uint	Unit of value
Usage	uint	
Value	Value	

7.8 Properties Service

7.8.1 PropertyNames

Property	Type	Description
ObjectId	string	Address of object to which value belongs to
Properties	Array of string	Array of property names corresponding to the ObjectId
FunctionProperties	Array of string	Array of function property names corresponding to the ObjectId

7.9 Command Service

7.9.1 PropertyCommand

Property	Type	Description
Commands	Array of Command	See Command [→ 66]
PropertyId	int	
ErrorCode	int	-ErrorCode >0, an error occurred. -ErrorCode = 0 , Success

7.9.2 Command

Property	Type	Description
_links	Array of Link	See Link [→ 58]
Descriptor	string	Localized command name
GroupNumber	int	If provided, commands with same GroupNumber belong together
Parameters	Array of CommandParameter	See CommandParameter [→ 66]
PropertyId	string	Address of property to which command belongs to
Id	string	Unique identifier of command

7.9.3 CommandParameters

Property	Type	Description
ControlType	ControlTypeId	See ControlTypeID [→ 67]
DataType	ApiDataType	
DefaultValue	string	Default value of parameter
Descriptor	string	Localized command parameter name
EnumerationTexts	Array of EnumItem	See EnumItem [→ 67]
Max	string	Maximum value if applicable
Min	string	Minimum value if applicable
Name	string	Unique identifier of command parameter within command
Order	int	Order of parameters within command

7.9.3.1 ControlTypeID

controlTyp*e* must have one of the following values.

controlType	Description
0	DropDown; control is required to show enumeration data.
1	Numeric; control is required to display numbers.
2	String; control is required to display text.
3	DateTime; control is required to display date and/or time.
4	Password; control is required to display a password.

7.9.4 EnumItem

Property	Type	Description
Descriptor	string	Localized value
Value	int	Numeric value (unique within enumeration)

7.10 Trend Service

7.10.1 TrendCollector

Property	Type	Description
CollectorObjectOrPropertyId	string	Address of collector object
ObjectId	string	Object ID of trended object
PropertyName	string	Trended property name
TrendseriesId	string	Identifier of corresponding trend series

7.10.2 TrendBorder

Property	Type	Description
From	DateTime	Minimum date/time from available values for requested trend series ID
To	DateTime	Maximum date/time from available values for requested trend series ID

7.10.3 TrendSeries

Property	Type	Description
Id	string	ID of trend series
Series	Array of Value	See Value [→ 59]
SeriesPropertyId	string	Trended object

7.11 Language Service

7.11.1 Language

Property	Type	Description
Descriptor	string	Full localized language name
Code	string	IETF language tag according to RFC5646 [5] [→ 12]

7.12 Table Service

7.12.1 SubDisciplines

Property	Type	Description
DisciplineId	Uint	Id of the Discipline
DisciplineDescriptor	string	Descriptor of the Discipline
SubDisciplines	Array of SubGroups	Array of SubDisciplines in SubGroups form for which filter is applied. See [→ 55]

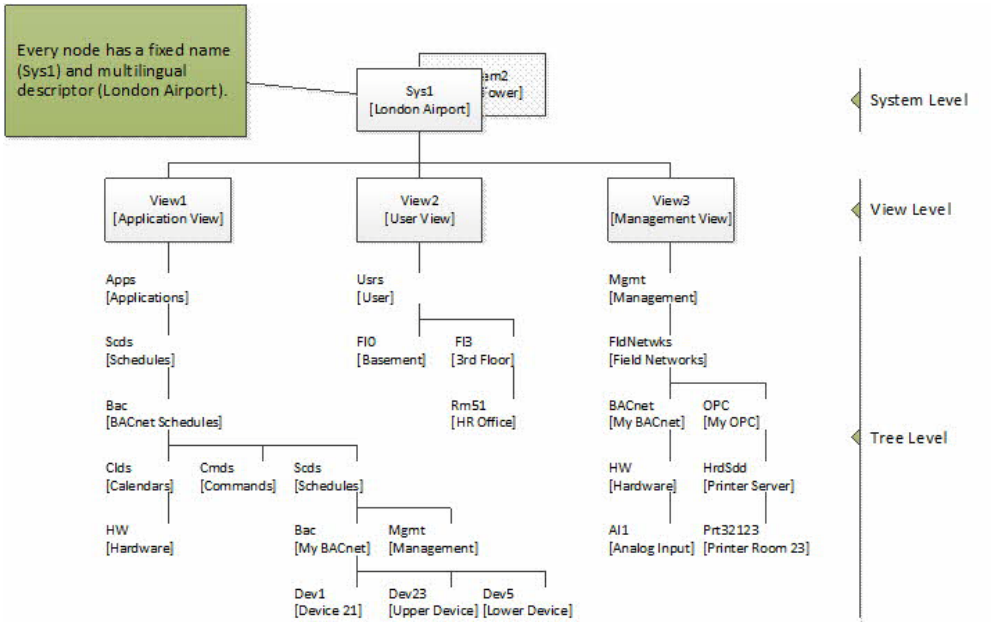
7.12.2 SubObjectTypes

Property	Type	Description
ObjectTypeId	Uint	Id of the ObjectType
ObjectTypeDescriptor	string	Descriptor of the ObjectType
SubObjectTypes	Array of SubGroups	Array of SubObjectTypes in SubGroups form for which filter is applied. See [→ 55]

8 Concepts

8.1 Naming

The following is an overview about data hierarchies and naming conventions in the system.



Name: The name is a fixed (untranslatable) literal of a node. In order to uniquely identify nodes deeper in the hierarchy, nodes can be concatenated. Further, to ensure uniqueness sibling nodes can never have the same name. Concatenated Names are called **Designation**.

```
<Designation> ::= <SystemName>.<ViewName>:{<TreeNodeName>}
```

Example of Designation (node **HW** [Hardware]):

```
Sys1.View1:Apps.Scds.Bac.Clds.HW
```

Descriptor: The descriptor is a multilingual text of a node. Although nodes deeper in the hierarchy can be concatenated, they DO NOT have to be unique! Concatenated Descriptors are called **Location**.

```
<Location> ::=
<SystemDescriptor>.<ViewDescriptor>:{<TreeNodeDescriptor>}
```

Example of Location (node **HW** [Hardware]):

```
London Airport.Application View:Applications.Schedules.BACnet
Schedules.Calendar.Hardware
```

```
public class SubExtensionsController : ApiController
{
    #region SignalR Hub

    readonly Lazy<IHubContext> _hub = new Lazy<IHubContext>(() =>
GlobalHost.ConnectionManager.GetHubContext<ExtensionsHub>());

    protected IHubContext Hub { /*omitted*/ }

    #endregion

    //send message to all clients
    public HttpResponseMessage Post(string id)
    {
        Hub.Clients.All.notifyMessage(id);
        return Request.CreateResponse(HttpStatusCode.OK);
    }
}
```

Issued by
Siemens Industry, Inc.
Building Technologies Division
1000 Deerfield Pkwy
Buffalo Grove IL 60089
Tel. +1 847-215-1000

Document ID A6V10438036_en_a_21
Edition 2015-07-15