SCADA Engine
11 Buccaneer Avenue,
Lammermoor, Qld 4703 Australia
Tel: (61) 0490 118 346
E-mail: sales@scadaengine.com

BACnet Simulator User's Manual

Contents

Introduction 3
Installation 4
System Requirements 4
Install 4
Installed Files 7
Application Data 7
Uninstall 7
Getting Started 8
Overview 8
Start the BACnet Simulator User Interface 8
Add BACnet device 9
Add BACnet Object 13
Update Object Properties 15
Updating a property with a simple value format 15
Updating a property with a value collection 17
Updating a property value with a choice 18
Non updatable properties 19
Alarms 19
Notification Class 19
Intrinsic Reporting 20
Algorithmic reporting 23
Site Backup 25
Project properties 26
Restore Site 27
Copy / Paste network objects 28
Delete Network Object 31
Exporting a Network or Device 33
Importing a network or device 34

Closing the application 36
Simulation settings 36
BACnet REST Web Server 38
Confirmed Service 39
Unconfirmed Service 39
Object Access 40
Customisation 41
Troubleshooting 42
Protocol Implementation Conformance Statement 42

Introduction

BACnet (Building Automation and Control Network) is a communication protocol for building automation and control systems. The BACnet simulator is a tool that allows users to test and simulate BACnet-based systems and devices. It allows developers to create and test BACnet applications, and it can also be used for training and education purposes.

The BACnet simulators can be used to simulate various types of BACnet devices, such as controllers, sensors, and actuators, and it can also simulate different BACnet network configurations and communication patterns. The BACnet simulator can be useful for testing the performance and functionality of BACnet systems and ensuring that they are working correctly before they are deployed in real-world environments.

The BACnet simulator uses a virtual network which is a software-based network that is created and simulated by the application. It is used to mimic the characteristics and behavior of a real-world BACnet network, such as a BACnet/MSTP or BACnet/IP network. Virtual networks are often used for testing and development purposes, as they allow users to create and experiment with different network configurations and scenarios without the need for physical hardware.

Within the BACnet virtual network 1 or more virtual devices can be created. A virtual device is a software-based representation of a physical BACnet device, such as a room controller. It can be used to simulate the behaviour and functionality of the physical device in a virtual environment. Virtual devices are often used in conjunction with virtual networks, as they allow users to create and test network configurations and communication patterns with virtual devices. They can be useful for training, development, and testing purposes, as they allow users to experiment with different configurations and scenarios without the need for physical hardware.

Installation

Installing the BACnet Simulator involves the following steps:

- Download the installation file for the application from the internet or obtain it on a physical medium such as a DVD.
- Navigate to the location where the installation file was saved and double-click on it to begin the installation process.
- Follow the on-screen prompts to complete the installation. This may include agreeing to a license agreement, selecting an installation location, and specifying any additional options or settings.
- If the application requires any additional components or dependencies to be installed, the installer may automatically download and install these as part of the process.
- Once the installation is complete, the application will be installed on your computer and will be ready to use. It may also be added to the Start menu or Start screen for easy access.
- Note: Some applications may require administrative privileges to install, in which case you may be prompted to enter the administrator password for your computer before the installation can proceed.

System Requirements

The minimum hardware requirements for the BACnet Simulator are:

- Intel® Pentium® 4 Processor
- 512 MB RAM
- 500 GB hard drive

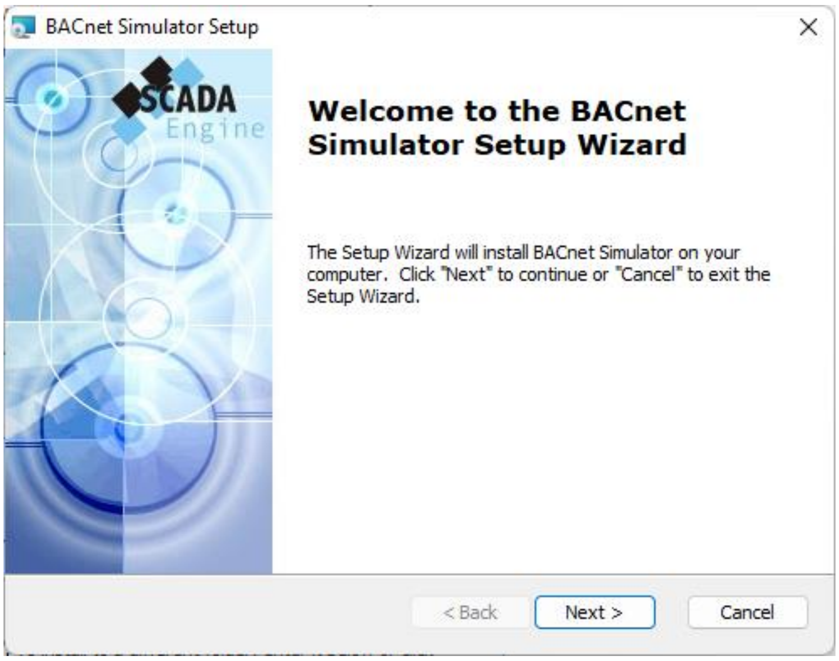
The CBMS studio BACnet Simulator can be used with the following operating systems:

- Microsoft Windows 7
- Microsoft Windows 2016 Server
- Microsoft Windows 10
- Microsoft Windows 11

Install

Log onto the system as Administrator before running the installation program, it cannot be installed under a limited user account.

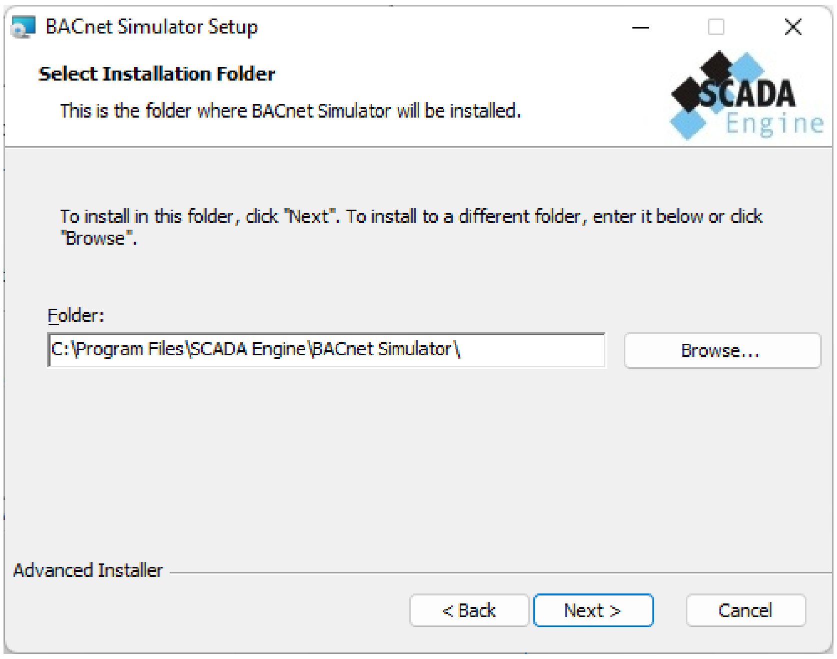
1. Place the CBMS studio BACnet Simulator CD into the CD drive, or double click on the ***BACnetSimulator*** program.
2. The Windows Installer will start and you should see the following screen.



Welcome to the BACnet Simulator Setup Wizard

The Setup Wizard will install BACnet Simulator on your computer. Click "Next" to continue or "Cancel" to exit the Setup Wizard.

3. Click the Next Button to select the installation folder.



Select Installation Folder

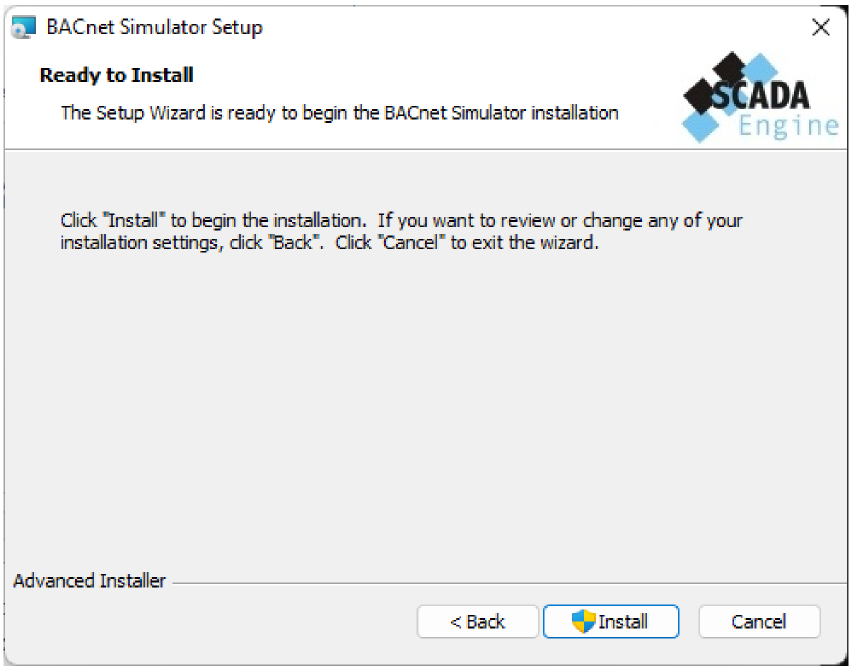
This is the folder where BACnet Simulator will be installed.

To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".

Folder: C:\Program Files\SCADA Engine\BACnet Simulator\

Browse for a new Installation folder or keep the default location.

4. Click next Button. You will be displayed the “Ready to Install” dialog.



BACnet Simulator Setup

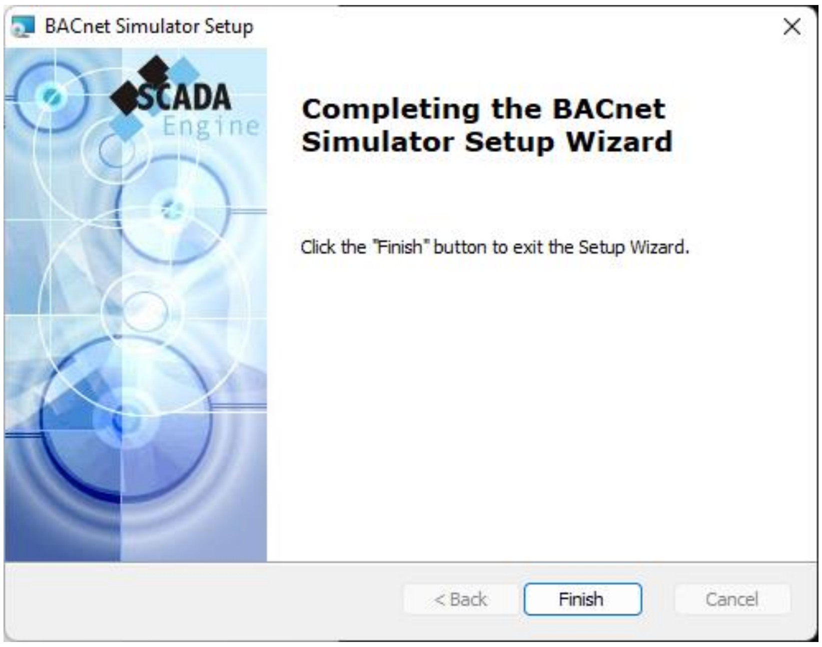
Ready to Install

The Setup Wizard is ready to begin the BACnet Simulator installation

Click "Install" to begin the installation. If you want to review or change any of your installation settings, click "Back". Click "Cancel" to exit the wizard.

Advanced Installer

5. Click the Install button and it will display the progress of the installation.
6. Once the installation is completed you will be displayed the following dialog. Click Finish Button to exit the wizard.



BACnet Simulator Setup

Completing the BACnet Simulator Setup Wizard

Click the "Finish" button to exit the Setup Wizard.

Installed Files

The program files are installed by default into the C:\Program Files\SCADA Engine\BACnet Simulator directory on the hard drive.

Application Data

Application data is stored onto the hard drive into the C:\Documents and Settings\All Users\Application Data\ SCADA Engine\BACnet Simulator directory for Windows 2000, 2003 and XP. It is stored into C:\ProgramData\ SCADA Engine\BACnet Simulator for Windows 10 and above.

Uninstall

To remove the BACnet Simulator, go to the Control Panel and select Add/Remove programs. Locate the entry for the BACnet Simulator and remove it.

Getting Started

This section contains a tutorial with a step by step walk through all the functionalities of the BACnet Simulator.

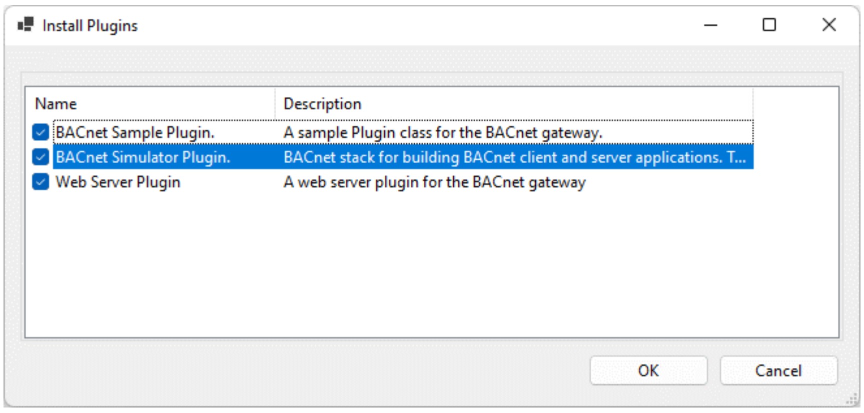
Overview

The simulator has a Windows service that runs in the background on the computer running the Windows operating system. It is designed to perform all the BACnet communication, and it can be configured to start automatically when the computer starts up. Unlike a regular application, a Windows service does not have a user interface that is visible to the user. Instead, it runs in the background and performs its tasks without requiring any input from the user.

A separate application includes a graphical user interface (GUI) that allows the user to interact with the service and configure its settings. This document describes the operation of the user interface which will be used to configure the windows service. The user interface can be closed at any time and the operation of the simulator will not be effected because the windows service will continue running.

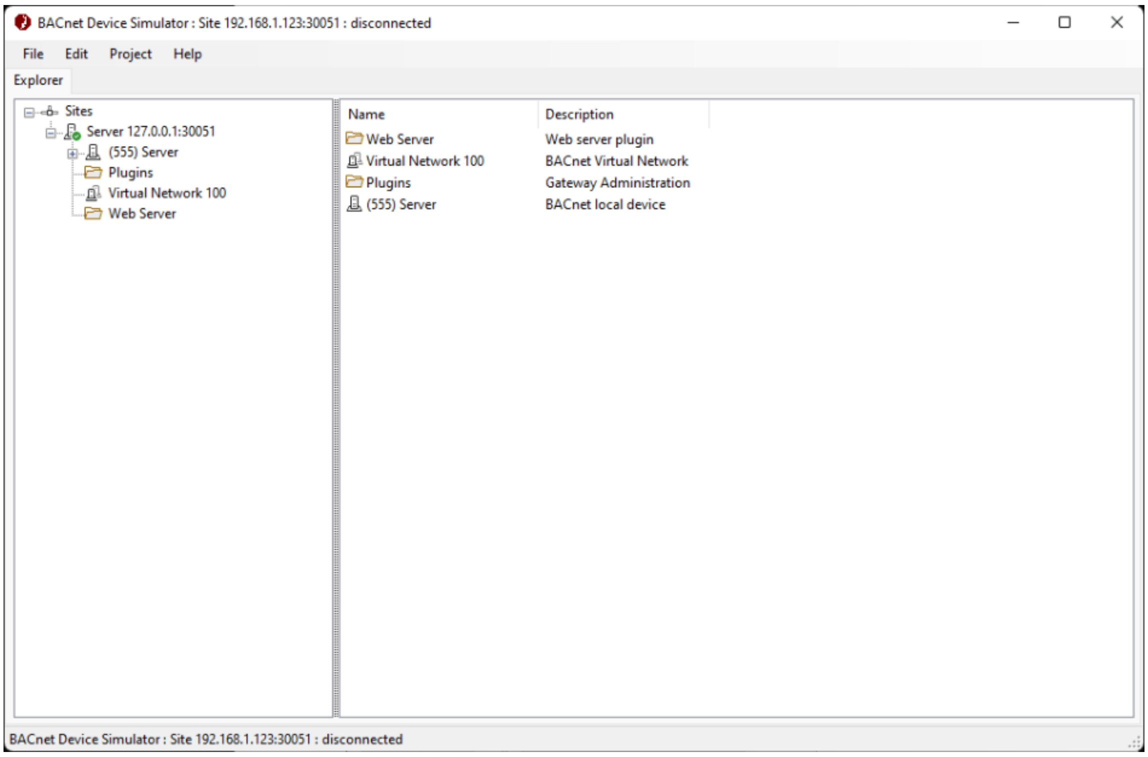
Start the BACnet Simulator User Interface

From the Start Menu select "***SCADA Engine -> BACnet Simulator***" to start the BACnet Simulator. You will see the following screen when you start it for the first time. Select the plugins that you wish to install and then select ok.



Name	Description
✓ BACnet Sample Plugin.	A sample Plugin class for the BACnet gateway.
✓ BACnet Simulator Plugin.	BACnet stack for building BACnet client and server applications. T...
✓ Web Server Plugin	A web server plugin for the BACnet gateway

You can then select the first node which corresponds to the simulator running on the local PC as shown in the diagram below.

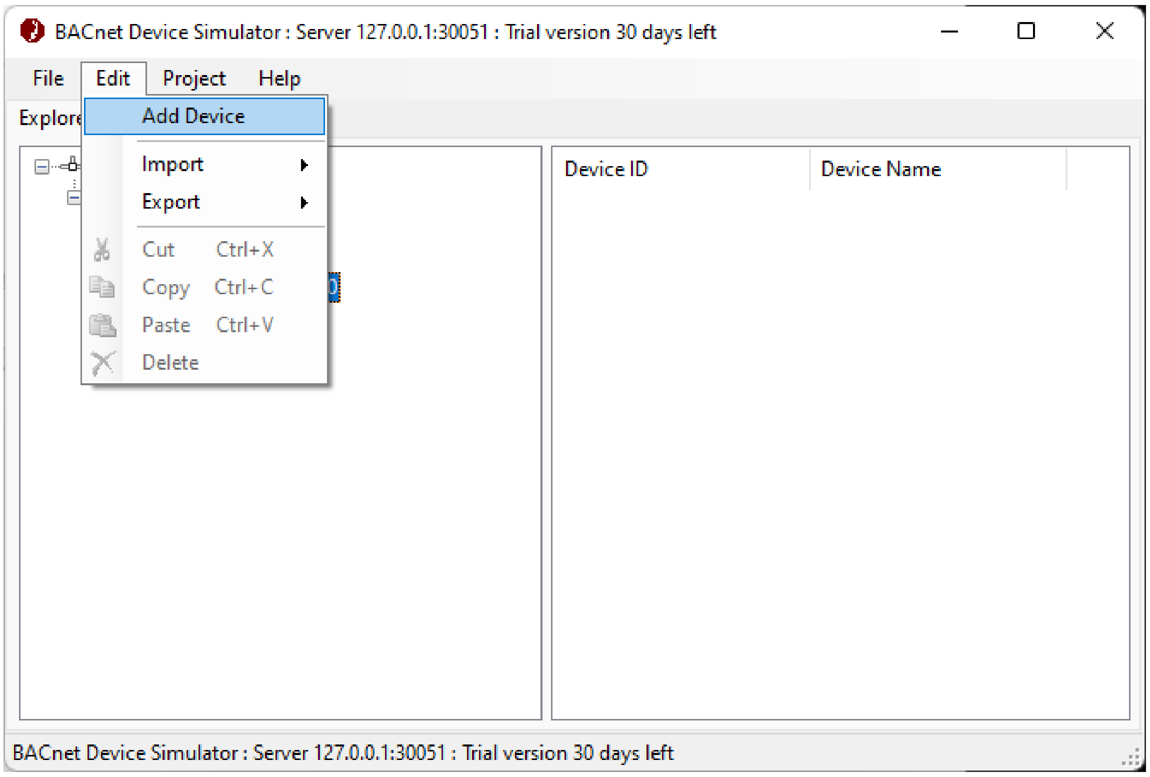


Name	Description
Web Server	Web server plugin
Virtual Network 100	BACnet Virtual Network
Plugins	Gateway Administration
(555) Server	BACnet local device

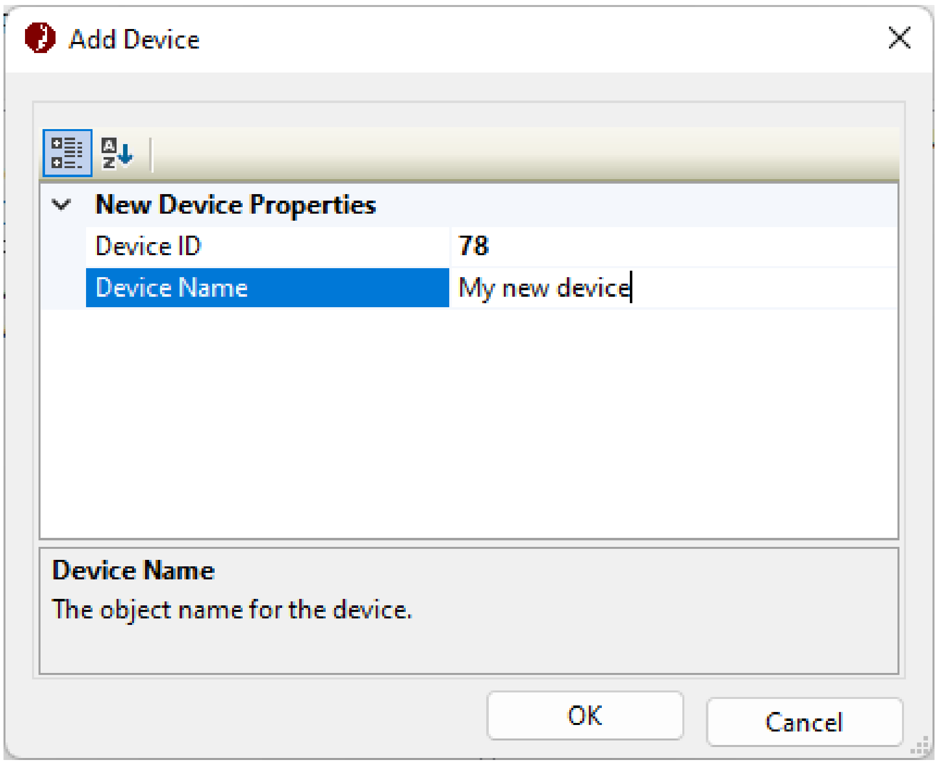
BACnet Device Simulator : Site 192.168.1.123:30051 : disconnected

Add BACnet device

You can add devices to the virtual network by navigating to the virtual network node and then selecting the ***Edit -> Add BACnet Device*** menu or by right clicking on the node and selecting Add ***BACnet Device.***


BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 30 days left
File
Edit
Project
Help
Explore
Add Device
Import
Export
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Delete
Device ID
Device Name
BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 30 days left

When you click on the **Add Device**, the following dialog box will be displayed.



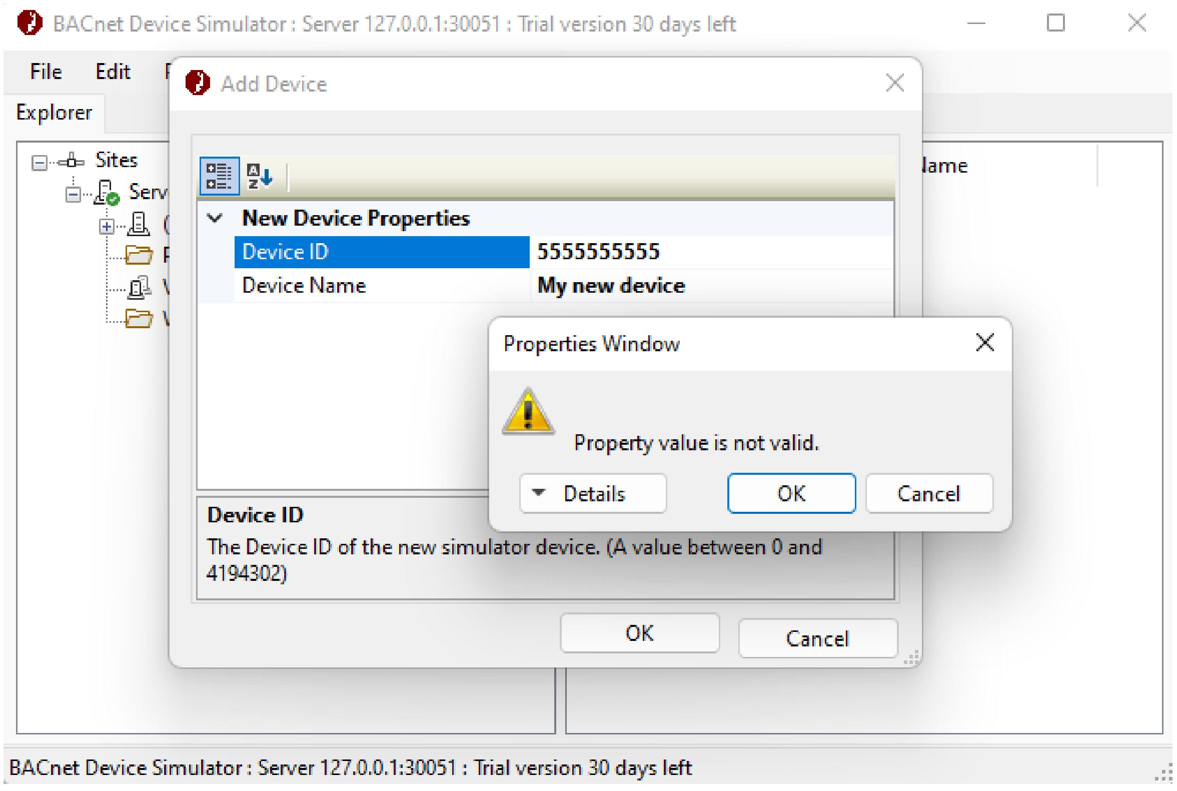
New Device Properties

Device ID	78
Device Name	My new device

Device Name
The object name for the device.

You must specify the Device ID and a name for the device. The device ID displayed is the next possible device ID. Users can change it as they need.

Device ID must be between 0 and 4194302. If user specifies an invalid ID it will display the following error message.


BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 30 days left
File
Edit
Explorer
Sites
Add Device
Details
OK
Cancel

New Device Properties

Device ID	5555555555
Device Name	My new device

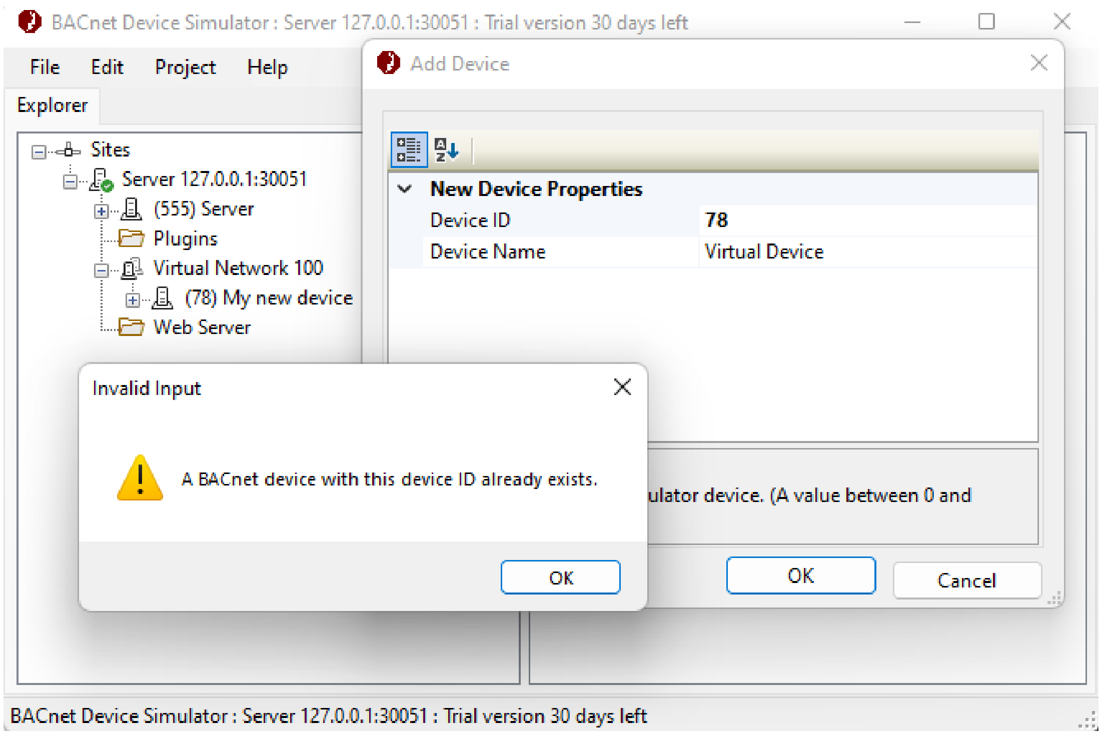
Properties Window

Property value is not valid.

Device ID
The Device ID of the new simulator device. (A value between 0 and 4194302)

BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 30 days left

If user enters an existing device ID, it will display the following error.



New Device Properties

Device ID	78
Device Name	Virtual Device

Invalid Input

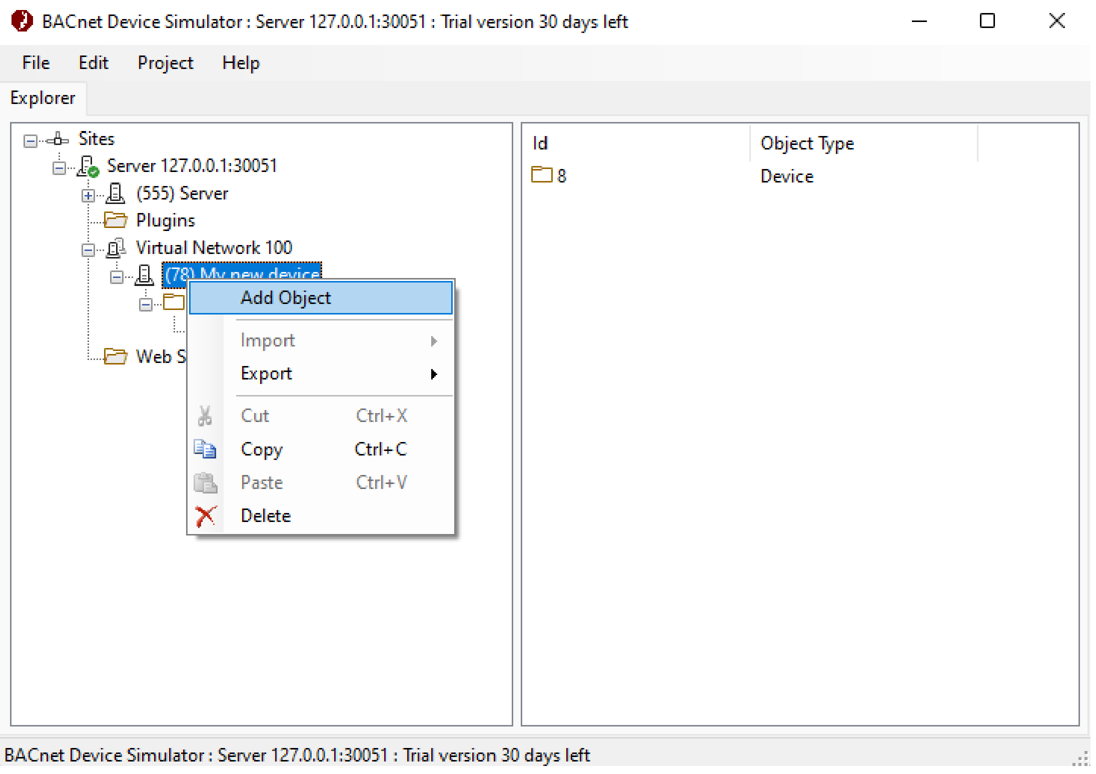
A BACnet device with this device ID already exists.

ulator device. (A value between 0 and

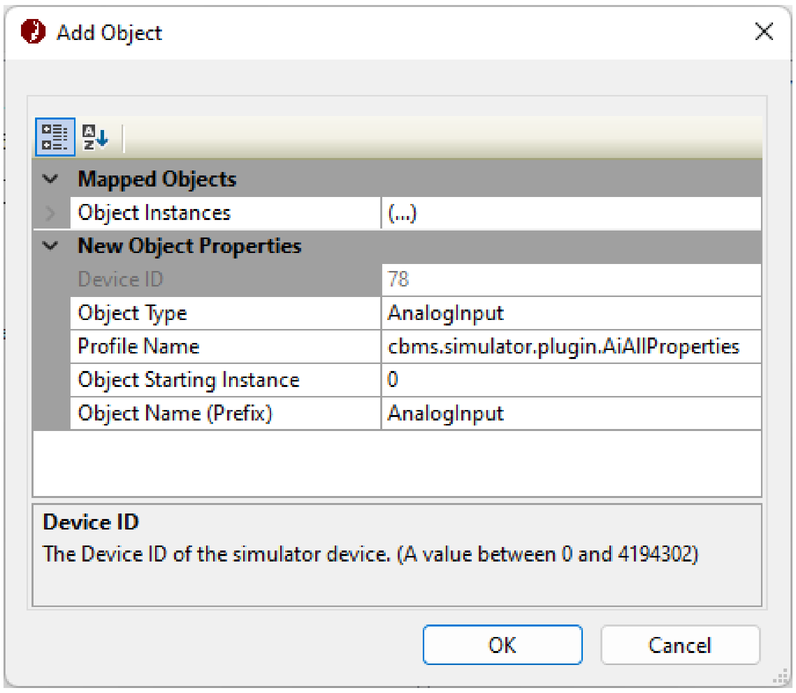
BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 30 days left

Add BACnet Object

Users can add BACnet objects to the selected device by using the menu ***Edit -> Add BACnet Object*** or ***Add Object*** in the popup menu which appears when you right click on the virtual device in the tree on the left panel.



When you click on the ***Add Object*** it will display the following dialog.



Mapped Objects

Object Instances	(...)

New Object Properties

Device ID	78
Object Type	AnalogInput
Profile Name	cbms.simulator.plugin.AiAllProperties
Object Starting Instance	0
Object Name (Prefix)	AnalogInput

Device ID
The Device ID of the simulator device. (A value between 0 and 4194302)

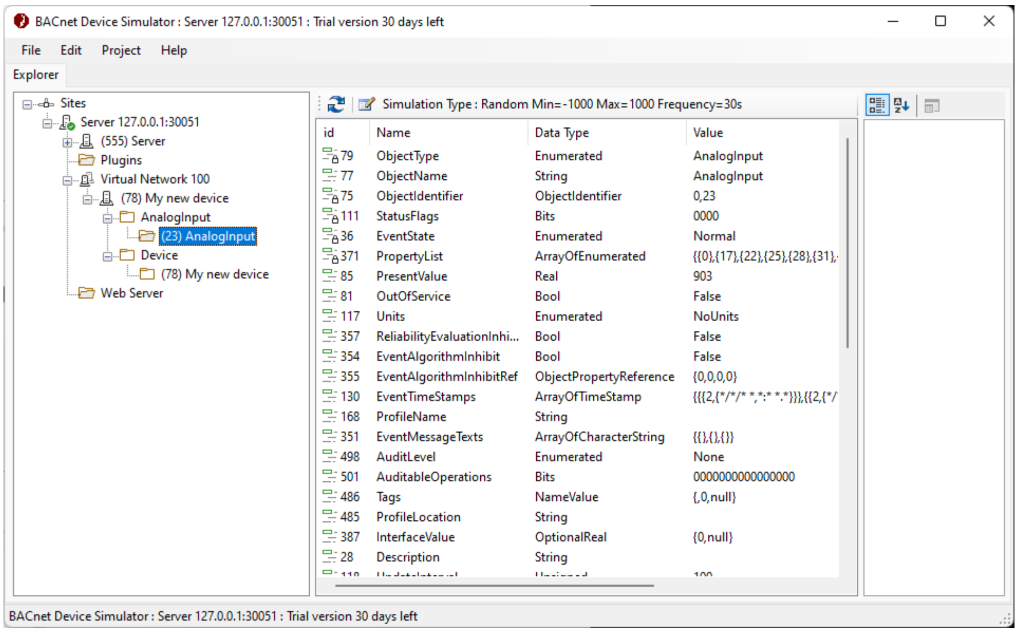
User has to specify the Device ID, Object Instance, Object Name, Object Type and Profile Name. Next possible Device ID, next Object Instance ID, Default object name and the selected object type will be displayed as the default values in the dialog. Users can change these values as they require.

Profiles have been created with different sets of properties within the object. Some BACnet properties are optional and may not appear in the object based on the profile that was selected.

Device ID and the Instance ID should be between 0 and 4194302. Object Type can be any of the available BACnet object types except the BACnetObjectTypeDevice.

If you are creating a new object of an existing object type, new object instance will be added under the existing object type folder. If it is a new object type, folder corresponds to the new object type will be created and an instance will be created under the newly created folder.

The devices, object types and instances hierarchy will be displayed like below.



Simulation Type : Random Min=-1000 Max=1000 Frequency=30s

id	Name	Data Type	Value
79	ObjectType	Enumerated	AnalogInput
77	ObjectName	String	AnalogInput
75	ObjectIdentifier	ObjectIdentifier	0,23
111	StatusFlags	Bits	0000
36	EventState	Enumerated	Normal
371	PropertyList	ArrayOfEnumerated	{{0},{17},{22},{25},{28},{31},
85	PresentValue	Real	903
81	OutOfService	Bool	False
117	Units	Enumerated	NoUnits
357	ReliabilityEvaluationInhi...	Bool	False
354	EventAlgorithmInhibit	Bool	False
355	EventAlgorithmInhibitRef	ObjectPropertyReference	{0,0,0,0}
130	EventTimeStamps	ArrayOfTimeStamp	{{{2,{*/*/* *,*:* *.*}}},{{2,{*/
168	ProfileName	String	
351	EventMessageTexts	ArrayOfCharacterString	{{},{},{}}
498	AuditLevel	Enumerated	None
501	AuditableOperations	Bits	0000000000000000
486	Tags	NameValue	{,0,null}
485	ProfileLocation	String	
387	InterfaceValue	OptionalReal	{0,null}
28	Description	String	

BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 30 days left

When you select a device, object type or an instance, details of the particular item will be displayed on the right hand pane.

If user selects the BACnet Network (root level) on the left hand tree, all the devices belongs to the network will be displayed on the right hand pane. If the selected item is a device, object types of that devices and if it is an object type, the instances of that object type will be displayed on the right had side. Also users can navigate further down by clicking the particular item in the right hand pane.

If you select an instance on the left hand tree panel or click on the instance name on the right hand panel, details of that instance will be displayed in the right hand panel.

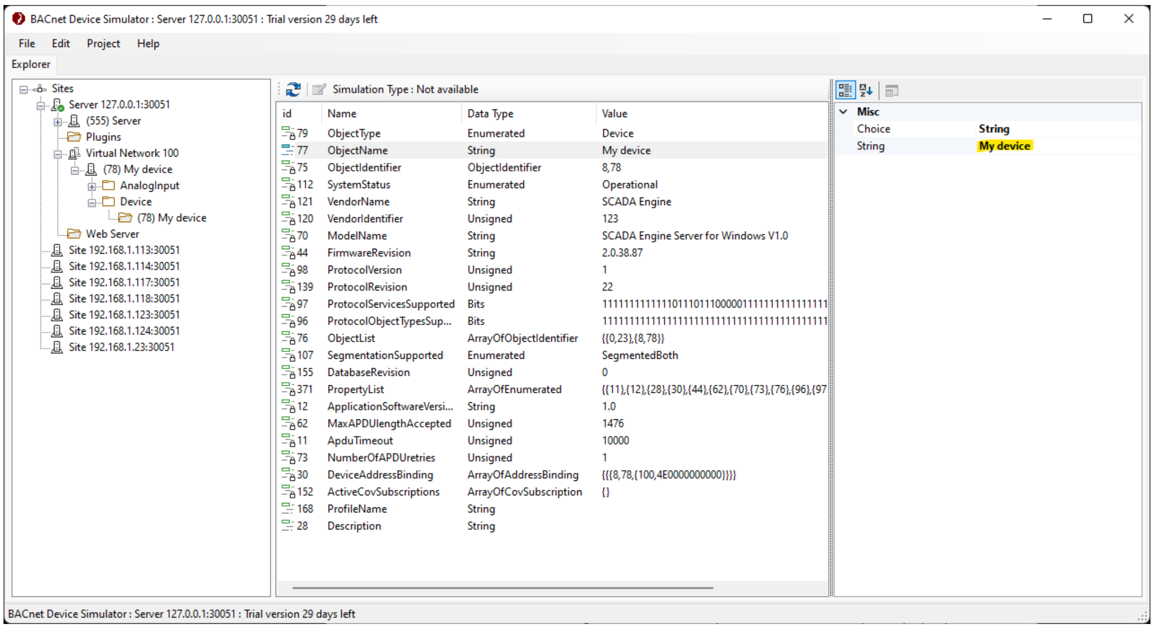
Update Object Properties

Users can update the object property values by clicking on the particular property value in the property grid. Read Only properties are displayed with a lock in the Icon and are not writable. For these properties the property grid is greyed to indicate that the value is read only.

The value can have different value formats such as simple value, selection from a collection of values, array, choices, etc. The method of displaying and updating the value depends on the value format.

Updating a property with a simple value format

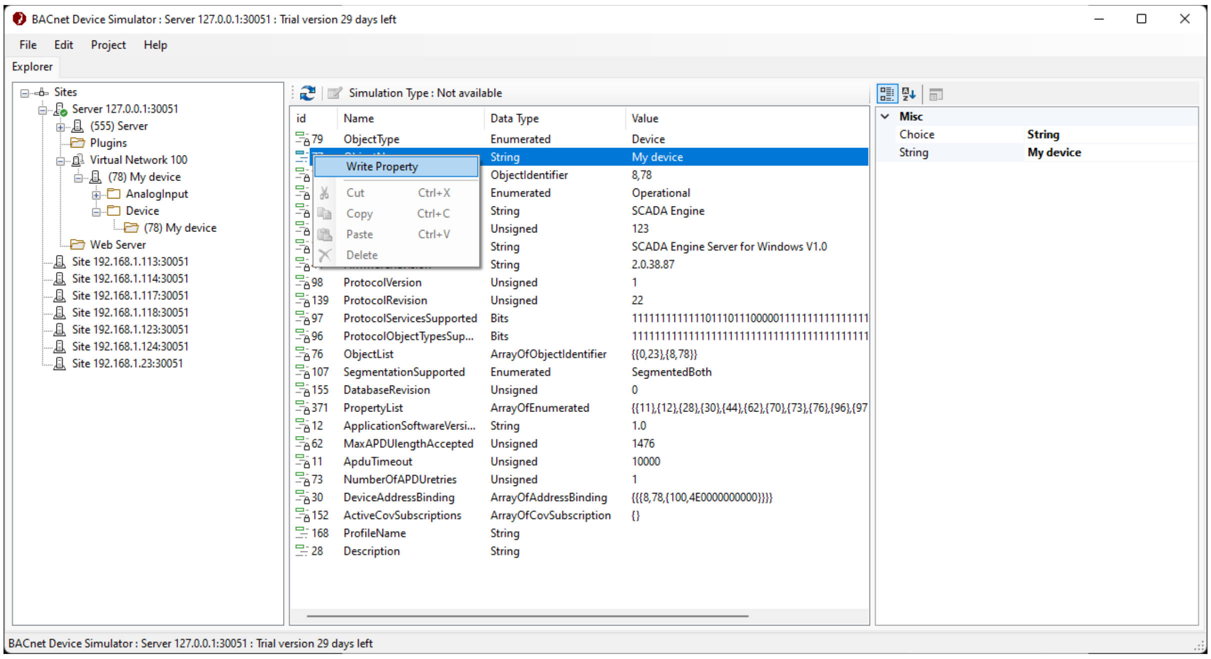
Select the corresponding value field in the right column of the property grid and simply edit the value in the property grid to the required value and press enter. For example, to change the object name select the property in the list and then change the text in the property grid.



id	Name	Data Type	Value
79	ObjectType	Enumerated	Device
77	ObjectName	String	My device
75	ObjectIdentifier	ObjectIdentifier	8,78
112	SystemStatus	Enumerated	Operational
121	VendorName	String	SCADA Engine
120	VendorIdentifier	Unsigned	123
70	ModelName	String	SCADA Engine Server for Windows V1.0
44	FirmwareRevision	String	2.0.38.87
98	ProtocolVersion	Unsigned	1
139	ProtocolRevision	Unsigned	22
97	ProtocolServicesSupported	Bits	111111111111101110111000001111111111111111
96	ProtocolObjectTypesSup...	Bits	111111111111111111111111111111111111111111
76	ObjectList	ArrayOfObjectIdentifier	{{0,23},{8,78}}
107	SegmentationSupported	Enumerated	SegmentedBoth
155	DatabaseRevision	Unsigned	0
371	PropertyList	ArrayOfEnumerated	{{11},{12},{28},{30},{44},{62},{70},{73},{76},{96},{97
12	ApplicationSoftwareVersi...	String	1.0
62	MaxAPDUlengthAccepted	Unsigned	1476
11	ApduTimeout	Unsigned	10000
73	NumberOfAPDUretries	Unsigned	1
30	DeviceAddressBinding	ArrayOfAddressBinding	{{{8,78,{100,4E0000000000}}}}
152	ActiveCovSubscriptions	ArrayOfCovSubscription	{}
168	ProfileName	String	
28	Description	String	

Choice	String
String	My device

The value can also be changed by a right click on the property and then selecting the **Write Property** menu item to bring up the write property dialog box.

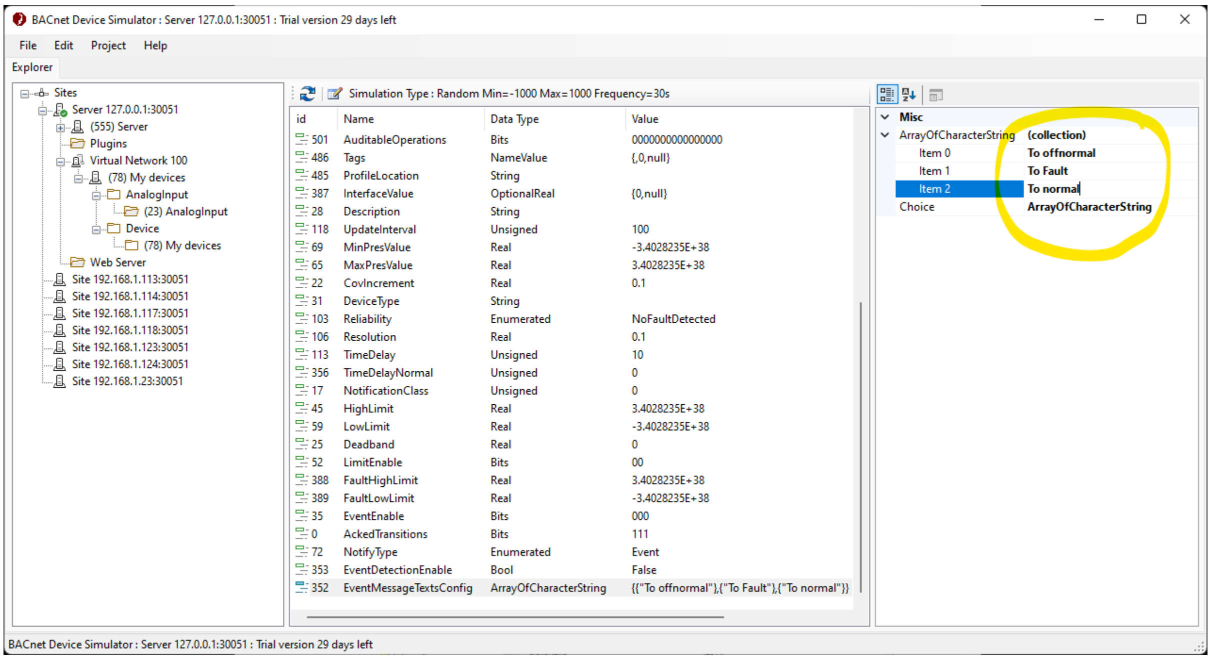


id	Name	Data Type	Value
79	ObjectType	Enumerated	Device
		String	My device
		ObjectIdentifier	8,78
		Enumerated	Operational
		String	SCADA Engine
		Unsigned	123
		String	SCADA Engine Server for Windows V1.0
		String	2.0.38.87
98	ProtocolVersion	Unsigned	1
139	ProtocolRevision	Unsigned	22
97	ProtocolServicesSupported	Bits	111111111111101110111000001111111111111111
96	ProtocolObjectTypesSup...	Bits	111111111111111111111111111111111111111111
76	ObjectList	ArrayOfObjectIdentifier	{{0,23},{8,78}}
107	SegmentationSupported	Enumerated	SegmentedBoth
155	DatabaseRevision	Unsigned	0
371	PropertyList	ArrayOfEnumerated	{{11},{12},{28},{30},{44},{62},{70},{73},{76},{96},{97
12	ApplicationSoftwareVersi...	String	1.0
62	MaxAPDUlengthAccepted	Unsigned	1476
11	ApduTimeout	Unsigned	10000
73	NumberOfAPDUretries	Unsigned	1
30	DeviceAddressBinding	ArrayOfAddressBinding	{{{8,78,{100,4E0000000000}}}}
152	ActiveCovSubscriptions	ArrayOfCovSubscription	{}
168	ProfileName	String	
28	Description	String	

Choice	String
String	My device

Updating a property with a value collection

If the property is a collection of values, it will display an expandable button (+) at the beginning of the property name. Users can expand and see the values of the sub properties and update them.


BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 29 days left

Simulation Type : Random Min=-1000 Max=1000 Frequency=30s

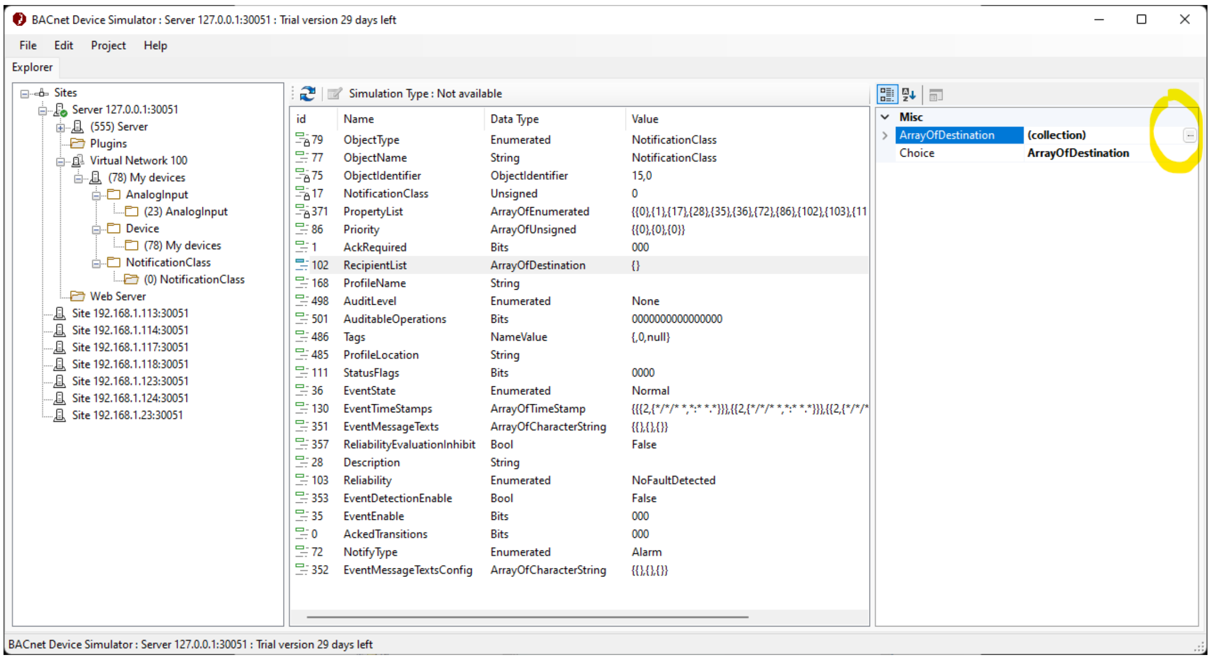
id	Name	Data Type	Value
501	AuditableOperations	Bits	0000000000000000
486	Tags	NameValue	{,0,null}
485	ProfileLocation	String	
387	InterfaceValue	OptionalReal	{0,null}
28	Description	String	
118	UpdateInterval	Unsigned	100
69	MinPresValue	Real	-3.4028235E+38
65	MaxPresValue	Real	3.4028235E+38
22	CovIncrement	Real	0.1
31	DeviceType	String	
103	Reliability	Enumerated	NoFaultDetected
106	Resolution	Real	0.1
113	TimeDelay	Unsigned	10
356	TimeDelayNormal	Unsigned	0
17	NotificationClass	Unsigned	0
45	HighLimit	Real	3.4028235E+38
59	LowLimit	Real	-3.4028235E+38
25	Deadband	Real	0
52	LimitEnable	Bits	00
388	FaultHighLimit	Real	3.4028235E+38
389	FaultLowLimit	Real	-3.4028235E+38
35	EventEnable	Bits	000
0	AckedTransitions	Bits	111
72	NotifyType	Enumerated	Event
353	EventDetectionEnable	Bool	False
352	EventMessageTextsConfig	ArrayOfCharacterString	{{"To offnormal"},{"To Fault"},{"To normal"}}

Misc	
ArrayOfCharacterString	(collection)
Item 0	To offnormal
Item 1	To Fault
Item 2	To normal
Choice	ArrayOfCharacterString

BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 29 days left

If the sub properties are simple values or selection values you can update them in a similar manner to the previous two sections.

Sometimes, the collection can hold an array of elements, an elipse button will appear to the right of this property and it will display a dialog box for add/removing items when it is selected.


BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 29 days left

File Edit Project Help

Explorer

- Sites
  - Server 127.0.0.1:30051
    - (555) Server
    - Plugins
    - Virtual Network 100
      - (78) My devices
        - AnalogInput
          - (23) AnalogInput
        - Device
          - (78) My devices
        - NotificationClass
          - (0) NotificationClass
    - Web Server
  - Site 192.168.1.113:30051
  - Site 192.168.1.114:30051
  - Site 192.168.1.117:30051
  - Site 192.168.1.118:30051
  - Site 192.168.1.123:30051
  - Site 192.168.1.124:30051
  - Site 192.168.1.23:30051

Simulation Type : Not available

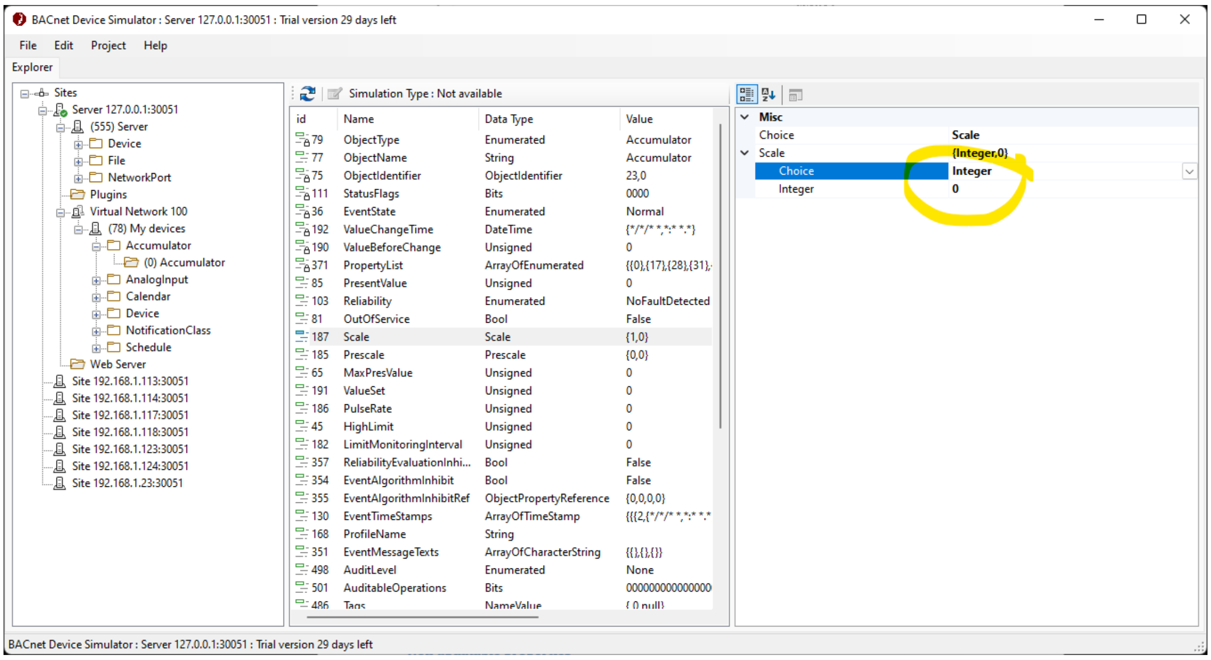
id	Name	Data Type	Value
79	ObjectType	Enumerated	NotificationClass
77	ObjectName	String	NotificationClass
75	ObjectIdentifier	ObjectIdentifier	15,0
17	NotificationClass	Unsigned	0
371	PropertyList	ArrayOfEnumerated	{{0},{1},{17},{28},{35},{36},{72},{86},{102},{103},{11
86	Priority	ArrayOfUnsigned	{{0},{0},{0}}
1	AckRequired	Bits	000
102	RecipientList	ArrayOfDestination	{}
168	ProfileName	String	
498	AuditLevel	Enumerated	None
501	AuditableOperations	Bits	0000000000000000
486	Tags	NameValue	{,0,null}
485	ProfileLocation	String	
111	StatusFlags	Bits	0000
36	EventState	Enumerated	Normal
130	EventTimeStamps	ArrayOfTimeStamp	{{{2,{*/*/* *,*:* *.*}}},{{2,{*/*/* *,*:* *.*}}},{{2,{*/*/*
351	EventMessageTexts	ArrayOfCharacterString	{{},{},{}}
357	ReliabilityEvaluationInhibit	Bool	False
28	Description	String	
103	Reliability	Enumerated	NoFaultDetected
353	EventDetectionEnable	Bool	False
35	EventEnable	Bits	000
0	AckedTransitions	Bits	000
72	NotifyType	Enumerated	Alarm
352	EventMessageTextsConfig	ArrayOfCharacterString	{{},{},{}}

Misc	
ArrayOfDestination	(collection)
Choice	ArrayOfDestination

BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 29 days left

Updating a property value with a choice

Some of the property values are choice values, which means that the displayed sub properties get changed based on the user's selection. The "choice" is a selection property from which you can select a value and based on that the other available sub properties get changed.


BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 29 days left

File Edit Project Help

Explorer

- Sites
  - Server 127.0.0.1:30051
    - (555) Server
      - Device
      - File
      - NetworkPort
    - Plugins
    - Virtual Network 100
      - (78) My devices
        - Accumulator
          - (0) Accumulator
        - AnalogInput
        - Calendar
        - Device
        - NotificationClass
        - Schedule
    - Web Server
  - Site 192.168.1.113:30051
  - Site 192.168.1.114:30051
  - Site 192.168.1.117:30051
  - Site 192.168.1.118:30051
  - Site 192.168.1.123:30051
  - Site 192.168.1.124:30051
  - Site 192.168.1.23:30051

Simulation Type : Not available

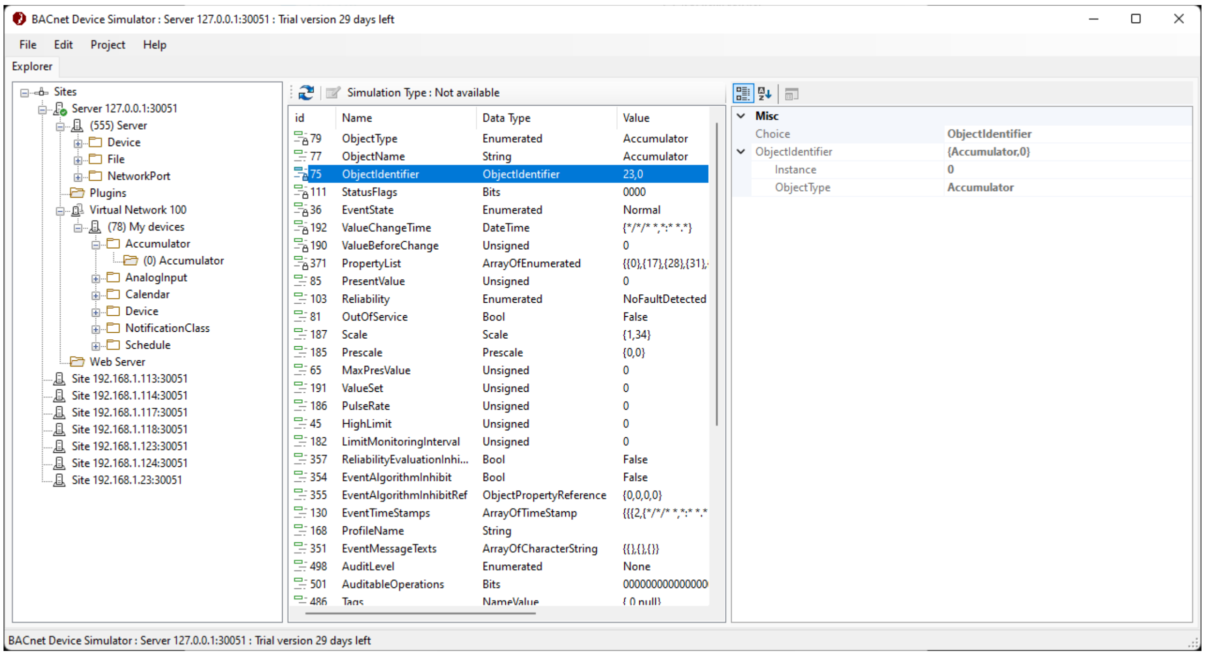
id	Name	Data Type	Value
79	ObjectType	Enumerated	Accumulator
77	ObjectName	String	Accumulator
75	ObjectIdentifier	ObjectIdentifier	23,0
111	StatusFlags	Bits	0000
36	EventState	Enumerated	Normal
192	ValueChangeTime	DateTime	{*/*/* *,*:* *.*}
190	ValueBeforeChange	Unsigned	0
371	PropertyList	ArrayOfEnumerated	{{0},{17},{28},{31},
85	PresentValue	Unsigned	0
103	Reliability	Enumerated	NoFaultDetected
81	OutOfService	Bool	False
187	Scale	Scale	{1,0}
185	Prescale	Prescale	{0,0}
65	MaxPresValue	Unsigned	0
191	ValueSet	Unsigned	0
186	PulseRate	Unsigned	0
45	HighLimit	Unsigned	0
182	LimitMonitoringInterval	Unsigned	0
357	ReliabilityEvaluationInhi...	Bool	False
354	EventAlgorithmInhibit	Bool	False
355	EventAlgorithmInhibitRef	ObjectPropertyReference	{0,0,0,0}
130	EventTimeStamps	ArrayOfTimeStamp	{{{2,{*/*/* *,*:* *.*
168	ProfileName	String	
351	EventMessageTexts	ArrayOfCharacterString	{{},{},{}}
498	AuditLevel	Enumerated	None
501	AuditableOperations	Bits	000000000000000
486	Tags	NameValue	{,0,null}

Misc	
Choice	Scale
Scale	{Integer,0}
Choice	Integer
Integer	0

BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 29 days left

Non updatable properties

Some properties are not allowed to be updated by the users and the property grid will appear greyed out and the values cannot be modified. The icon has a lock symbol to indicate that these are read only.



id	Name	Data Type	Value
79	ObjectType	Enumerated	Accumulator
77	ObjectName	String	Accumulator
75	ObjectIdentifier	ObjectIdentifier	23,0
111	StatusFlags	Bits	0000
36	EventState	Enumerated	Normal
192	ValueChangeTime	DateTime	{*/*/* *,*:* *.*}
190	ValueBeforeChange	Unsigned	0
371	PropertyList	ArrayOfEnumerated	{{0},{17},{28},{31},
85	PresentValue	Unsigned	0
103	Reliability	Enumerated	NoFaultDetected
81	OutOfService	Bool	False
187	Scale	Scale	{1,34}
185	Prescale	Prescale	{0,0}
65	MaxPresValue	Unsigned	0
191	ValueSet	Unsigned	0
186	PulseRate	Unsigned	0
45	HighLimit	Unsigned	0
182	LimitMonitoringInterval	Unsigned	0
357	ReliabilityEvaluationInhi...	Bool	False
354	EventAlgorithmInhibit	Bool	False
355	EventAlgorithmInhibitRef	ObjectPropertyReference	{0,0,0,0}
130	EventTimeStamps	ArrayOfTimeStamp	{{{2,{*/*/* *,*:* *.*
168	ProfileName	String	
351	EventMessageTexts	ArrayOfCharacterString	{{},{},{}}
498	AuditLevel	Enumerated	None
501	AuditableOperations	Bits	000000000000000
486	Tags	NameValue	{ 0 null}

Misc	
Choice	ObjectIdentifier
ObjectIdentifier	{Accumulator,0}
Instance	0
ObjectType	Accumulator

BACnet Device Simulator : Server 127.0.0.1:30051 : Trial version 29 days left

Alarms

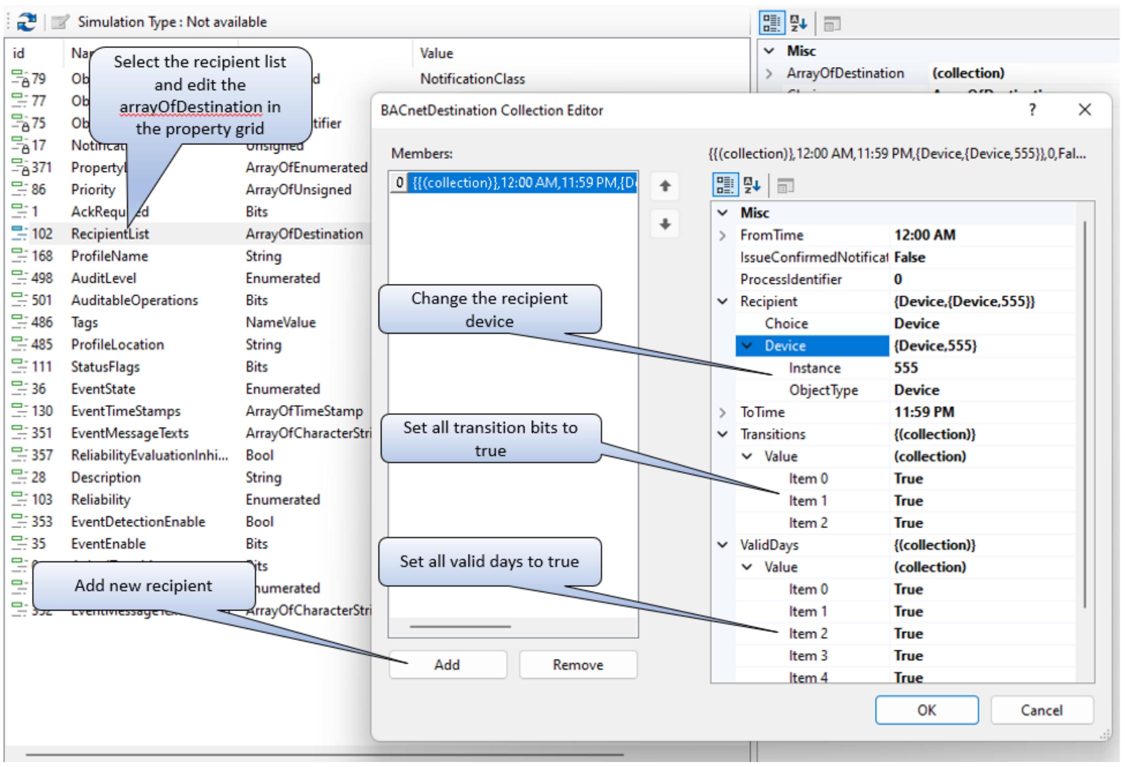
The OPC Simulator supports both Intrinsic reporting and Algorithmic reporting for generating alarms and sending them to other BACnet devices. Alarm generating objects within the simulator will trigger an alarm using the standard BACnet Alarm algorithms, for example a Binary Input can trigger an alarm when the presentValue equals the alarmValue. When an alarm is activate the simulator checks it's corresponding notification class object and send the alarm to the recipient.

Notification Class

In order to send alarms there must be one or more Notification Class objects configured within the same Virtual device of the simulator. Each Virtual device maintains its own independent Notification Class object. When an alarm occurs the simulator searches for its corresponding Notification Class object and then send the alarms to all the recipients in the list. To configure the notification class object, the following properties must be set.

- ackRequired – This property indicates if the alarm requires an acknowledgment.
- recipientList – This is a list of all recipients which will receive the alarm.

In order to send an alarm to a BACnet Workstation or any other BACnet device that will receive the alarm there must be an entry within the recipientList. In the simulator a recipient can be added and then all flags within the validDays and transitions properties should be set to try. The from time must begin at 12 am and the to time at 11:59 pm the issueConfirmedNotifications property can be set to true. The recipient type should be device and the device id of the operator workstation should be entered here. The screenshot below shows how to set all of the properties so that device 555 will receive alarms.


Select the recipient list and edit the arrayOfDestination in the property grid
Change the recipient device
Set all transition bits to true
Set all valid days to true
Add new recipient

Simulation Type : Not available

id	Name	Value
79	Ob	
77	Ob	NotificationClass
75	Ob	
17	Notificat	Unsigned
371	Property	ArrayOfEnumerated
86	Priority	ArrayOfUnsigned
1	AckRequired	Bits
102	RecipientList	ArrayOfDestination
168	ProfileName	String
498	AuditLevel	Enumerated
501	AuditableOperations	Bits
486	Tags	NameValue
485	ProfileLocation	String
111	StatusFlags	Bits
36	EventState	Enumerated
130	EventTimeStamps	ArrayOfTimeStamp
351	EventMessageTexts	ArrayOfCharacterStri
357	ReliabilityEvaluationInhi...	Bool
28	Description	String
103	Reliability	Enumerated
353	EventDetectionEnable	Bool
35	EventEnable	Bits

Misc
ArrayOfDestination (collection)

BACnetDestination Collection Editor

Members:
0 {{(collection)},12:00 AM,11:59 PM,{D

{{(collection)},12:00 AM,11:59 PM,{Device,{Device,555}},0,Fal...

Misc	
FromTime	12:00 AM
IssueConfirmedNotificat	False
ProcessIdentifier	0
Recipient	{Device,{Device,555}}
Choice	Device
Device	{Device,555}
Instance	555
ObjectType	Device
ToTime	11:59 PM
Transitions	{(collection)}
Value	(collection)
Item 0	True
Item 1	True
Item 2	True
ValidDays	{(collection)}
Value	(collection)
Item 0	True
Item 1	True
Item 2	True
Item 3	True
Item 4	True

Add Remove OK Cancel

Intrinsic Reporting

Intrinsic reporting is the term used when an object is capable of generating an alarm based on a set of properties associated with the object. It is a convenient way to configure alarms on BACnet points within the same device.

Intrinsic reporting is not supported on all object type, however all properties are available for reading and writing. The simulator supports intrinsic reporting for the following objects

- AnalogInput
- AnalogOutput
- AnalogValue
- BinaryInput
- BinaryOutput
- BinaryValue
- MultistateInput

- MultistateOutput
- MultistateValue

To configure the class object, the following properties must be set.

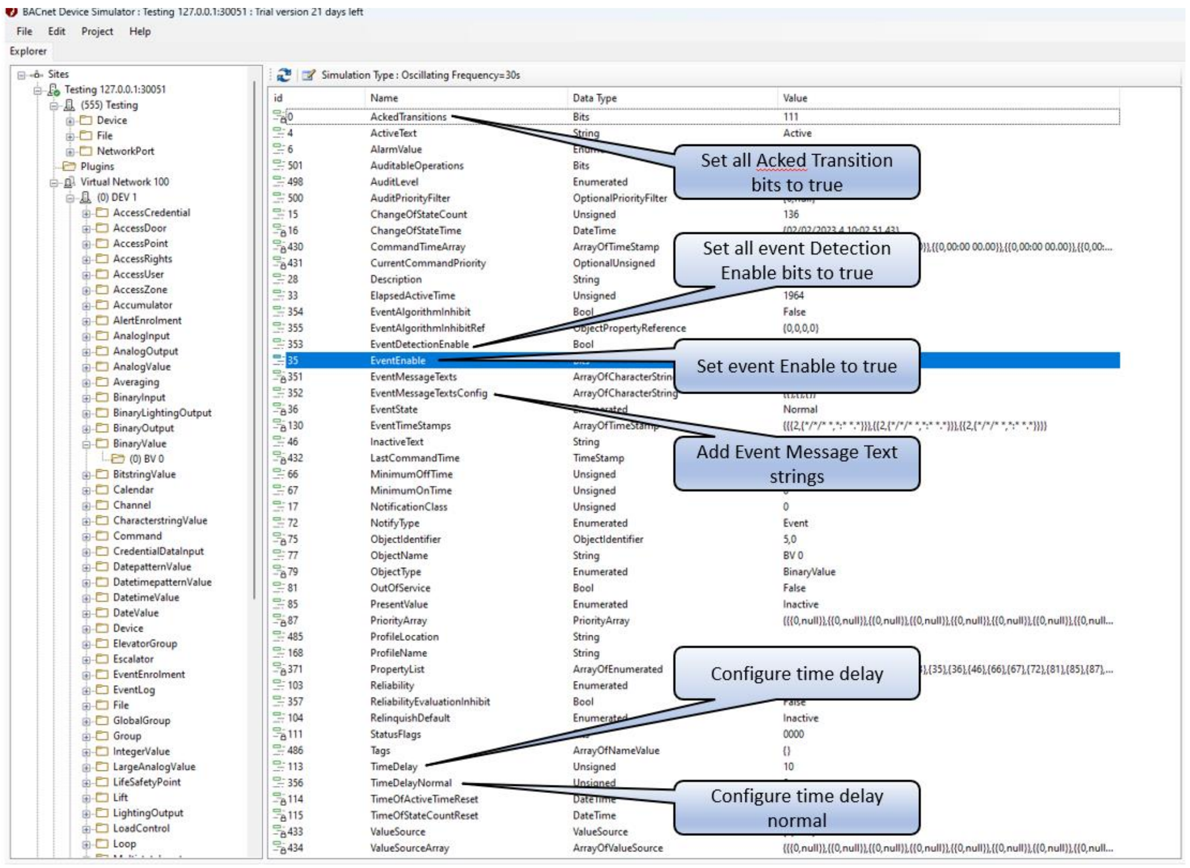
- eventDetectionEnable – This property must be set to true in order to generate alarms.
- eventEnable – Set all flags to true to generate all types of alarms.
- eventMessageTextsConfig – These are the messages that will be sent when an alarm is generated.
- notificationClass – This is reference to a notificationClass object within the same device which contains a list of recipients where the alarm will be sent.
- timeDelay – The time delay before an offNormal alarm is generated.
- timeDelayNormal – The time delay before a toNormal alarm is generated.

Object specific properties – Each object type has additional properties used by the alarm algorithm for generating alarms. For example alarmValue property of the BinaryValue object is used as a comparison against the presentValue.

After an alarm is generated the eventStatus will transition to offNormal and the eventMessageTexts property will contain the message to be transmitted with the alarm. Several other properties will also change state including the eventTimeStamps and ackedTransitions.

It is best to use the AnalogValue, BinaryValue and BinaryOutput objects when generating alarms manually. For example to activate the alarm for a BinaryValue object simply set the presentValue to the active state an alarm will be generated.

The screenshot below is an example of how a BinaryValue object can be used to generate an alarm.



id	Name	Data Type	Value
0	AckedTransitions	Bits	111
4	ActiveText	String	Active
6	AlarmValue	[illegible]	
501	AuditableOperations	Bits	
498	AuditLevel	Enumerated	
500	AuditPriorityFilter	OptionalPriorityFilter	[illegible]
15	ChangeOfStateCount	Unsigned	136
16	ChangeOfStateTime	DateTime	{02/02/2023 4 10:02 51 43}
430	CommandTimeArray	ArrayOfTimeStamp	}},{{0,00:00 00.00}},{{0,00:00 00.00}},{{0,00:...
431	CurrentCommandPriority	OptionalUnsigned	
28	Description	String	
33	ElapsedActiveTime	Unsigned	1964
354	EventAlgorithmInhibit	Bool	False
355	EventAlgorithmInhibitRef	ObjectPropertyReference	{0,0,0,0}
353	EventDetectionEnable	Bool	
35	EventEnable	[illegible]	
351	EventMessageTexts	ArrayOfCharacterString	
352	EventMessageTextsConfig	ArrayOfCharacterString	[illegible]
36	EventState	Enumerated	Normal
130	EventTimeStamps	ArrayOfTimeStamp	{{{2,{*/*/* *,*:* *.*}}},{{2,{*/*/* *,*:* *.*}}},{{2,{*/*/* *,*:* *.*}}}}
46	InactiveText	String	
432	LastCommandTime	TimeStamp	
66	MinimumOffTime	Unsigned	
67	MinimumOnTime	Unsigned	[illegible]
17	NotificationClass	Unsigned	0
72	NotifyType	Enumerated	Event
75	ObjectIdentifier	ObjectIdentifier	5,0
77	ObjectName	String	BV 0
79	ObjectType	Enumerated	BinaryValue
81	OutOfService	Bool	False
85	PresentValue	Enumerated	Inactive
87	PriorityArray	PriorityArray	{{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null...
485	ProfileLocation	String	
168	ProfileName	String	
371	PropertyList	ArrayOfEnumerated	},{35},{36},{46},{66},{67},{72},{81},{85},{87},...
103	Reliability	Enumerated	
357	ReliabilityEvaluationInhibit	Bool	False
104	RelinquishDefault	Enumerated	Inactive
111	StatusFlags	[illegible]	0000
486	Tags	ArrayOfNameValue	{}
113	TimeDelay	Unsigned	10
356	TimeDelayNormal	Unsigned	[illegible]
114	TimeOfActiveTimeReset	DateTime	
115	TimeOfStateCountReset	DateTime	
433	ValueSource	ValueSource	[illegible]
434	ValueSourceArray	ArrayOfValueSource	{{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null}},{{0,null...

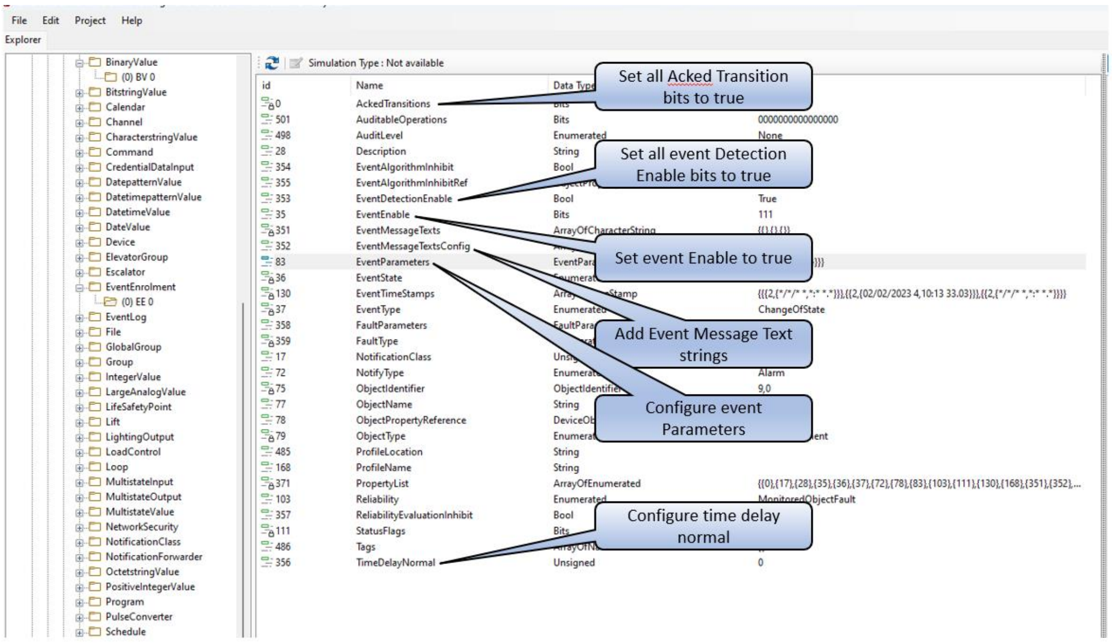
Algorithmic reporting

The simulator supports algorithmic reporting using the eventEnrolment object. There are many possible alogirithms that are available in bacnet for generating an alarm, for example change of state is used to generate an alarm on a BinaryInput object. This type of alarming is most often used when a device does not support alarms and an alarm is required to be generated. The eventEnrolment object will read the referenced property and based on the algoritm it will generate an alarm.

To configure the class object, the following properties must be set.

- eventDetectionEnable – This property must be set to true in order to generate alarms.
- eventEnable – Set all flags to true to generate all types of alarms.
- eventMessageTextsConfig – These are the messages that will be sent when an alarm is generated.
- eventParameters – This property contains all of the parameters used for generating the offNormal alarm.
- faultParameters – This property is used by the fault algorithm.
- objectPropertyReference – This is the reference to the value used to generate an alarm.
- notificationClass – This is reference to a notificationClass object within the same device which contains a list of recipients where the alarm will be sent.
- timeDelay – The time delay before an offNormal alarm is generated.
- timeDelayNormal – The time delay before a toNormal alarm is generated.

The screenshot below provides an example for generating an alarm when BinaryValue 0 changes state.

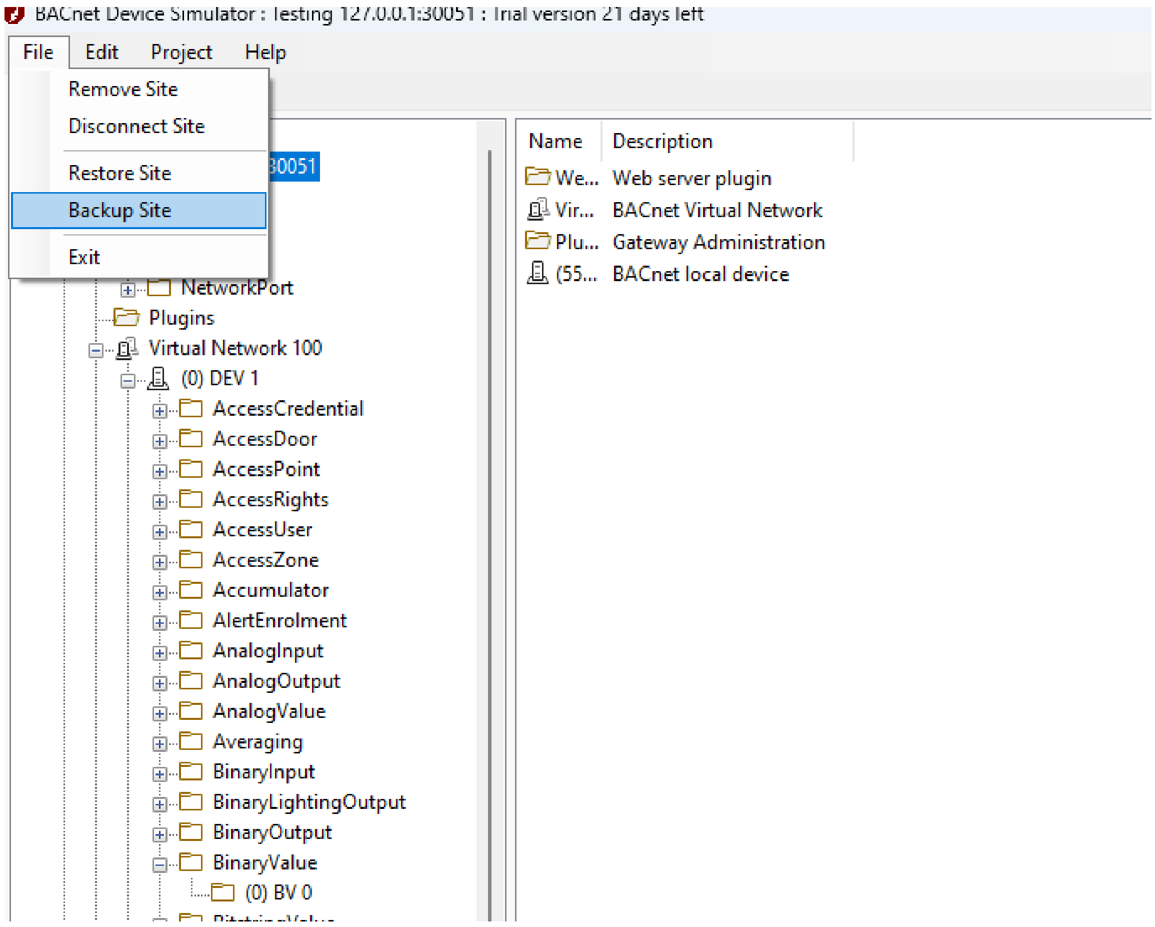


id	Name	Data Type	
0	AckedTransitions	Bits	
501	AuditableOperations	Bits	0000000000000000
498	AuditLevel	Enumerated	None
28	Description	String	
354	EventAlgorithmInhibit	Bool	
355	EventAlgorithmInhibitRef	ObjectPro	
353	EventDetectionEnable	Bool	True
35	EventEnable	Bits	111
351	EventMessageTexts	ArrayOfCharacterString	{{},{},{}}
352	EventMessageTextsConfig	Array	
83	EventParameters	EventPara	}}}
36	EventState	Enumerat	
130	EventTimeStamps	Array Stamp	{{{2,{*/*/* *,*:* *.*}}},{{2,{02/02/2023 4,10:13 33.03}}},{{2,{*/*/* *,*:* *.*}}}}
37	EventType	Enumerated	ChangeOfState
358	FaultParameters	FaultPara	
359	FaultType	Enumerat	
17	NotificationClass	Unsig	
72	NotifyType	Enumerat	Alarm
75	ObjectIdentifier	ObjectIdentifier	9,0
77	ObjectName	String	
78	ObjectPropertyReference	DeviceOb	
79	ObjectType	Enumerat	ent
485	ProfileLocation	String	
168	ProfileName	String	
371	PropertyList	ArrayOfEnumerated	{{0},{17},{28},{35},{36},{37},{72},{78},{83},{103},{111},{130},{168},{351},{352},...
103	Reliability	Enumerated	MonitoredObjectFault
357	ReliabilityEvaluationInhibit	Bool	
111	StatusFlags	Bits	
486	Tags	ArrayOfN	
356	TimeDelayNormal	Unsigned	0

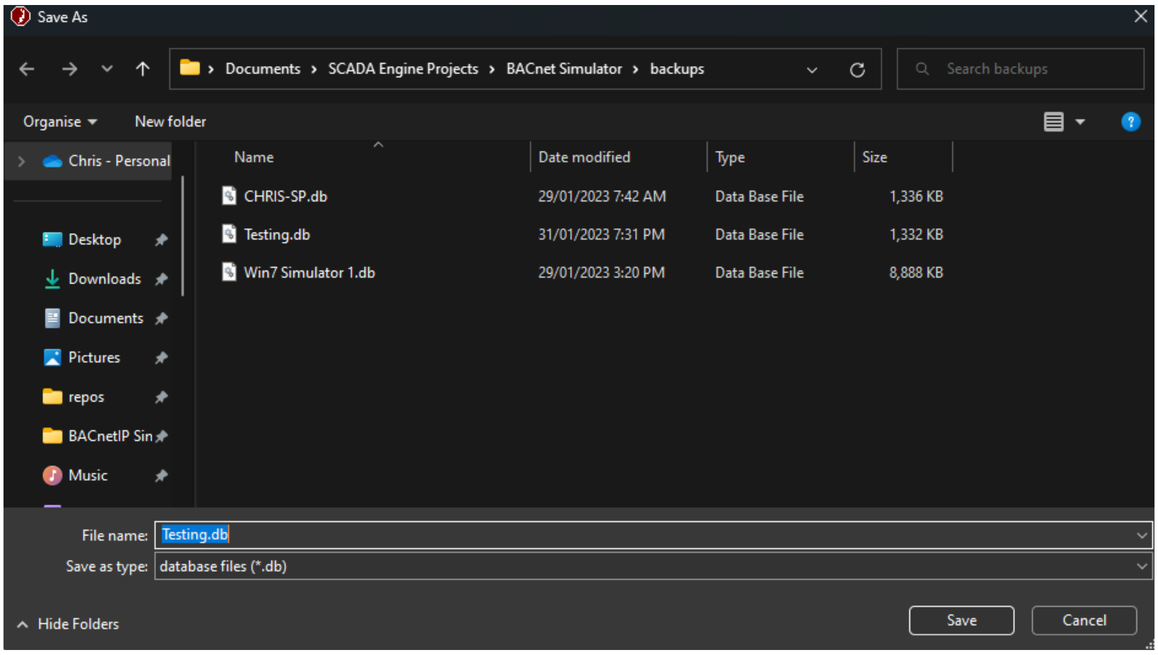
SCADA Engine
11 Buccaneer Avenue,
Lammermoor, Qld 4703 Australia
Tel: (61) 0490 118 346
E-mail: sales@scadaengine.com

Site Backup

Users can backup the database to a .db file by selecting the site node and then clicking the ***File->Backup Site*** menu.



When you click on one of the above menus, File Save dialog will appear and users can specify the file name and the location for the new file. The site will be saved as a .db file.

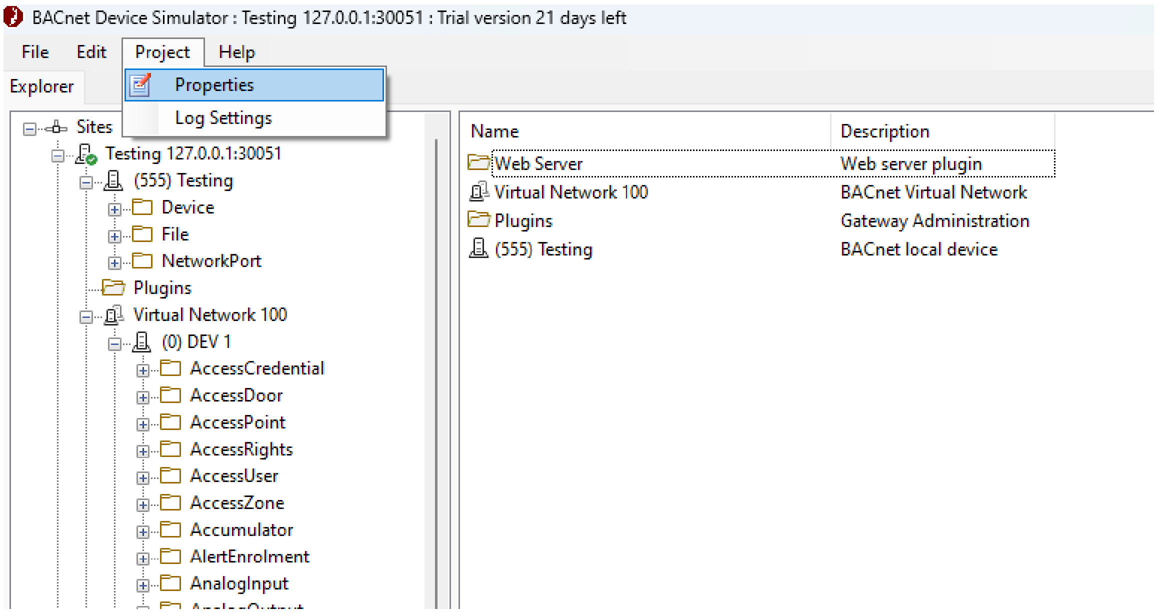


Name	Date modified	Type	Size
CHRIS-SP.db	29/01/2023 7:42 AM	Data Base File	1,336 KB
Testing.db	31/01/2023 7:31 PM	Data Base File	1,332 KB
Win7 Simulator 1.db	29/01/2023 3:20 PM	Data Base File	8,888 KB

File name: Testing.db
Save as type: database files (*.db)

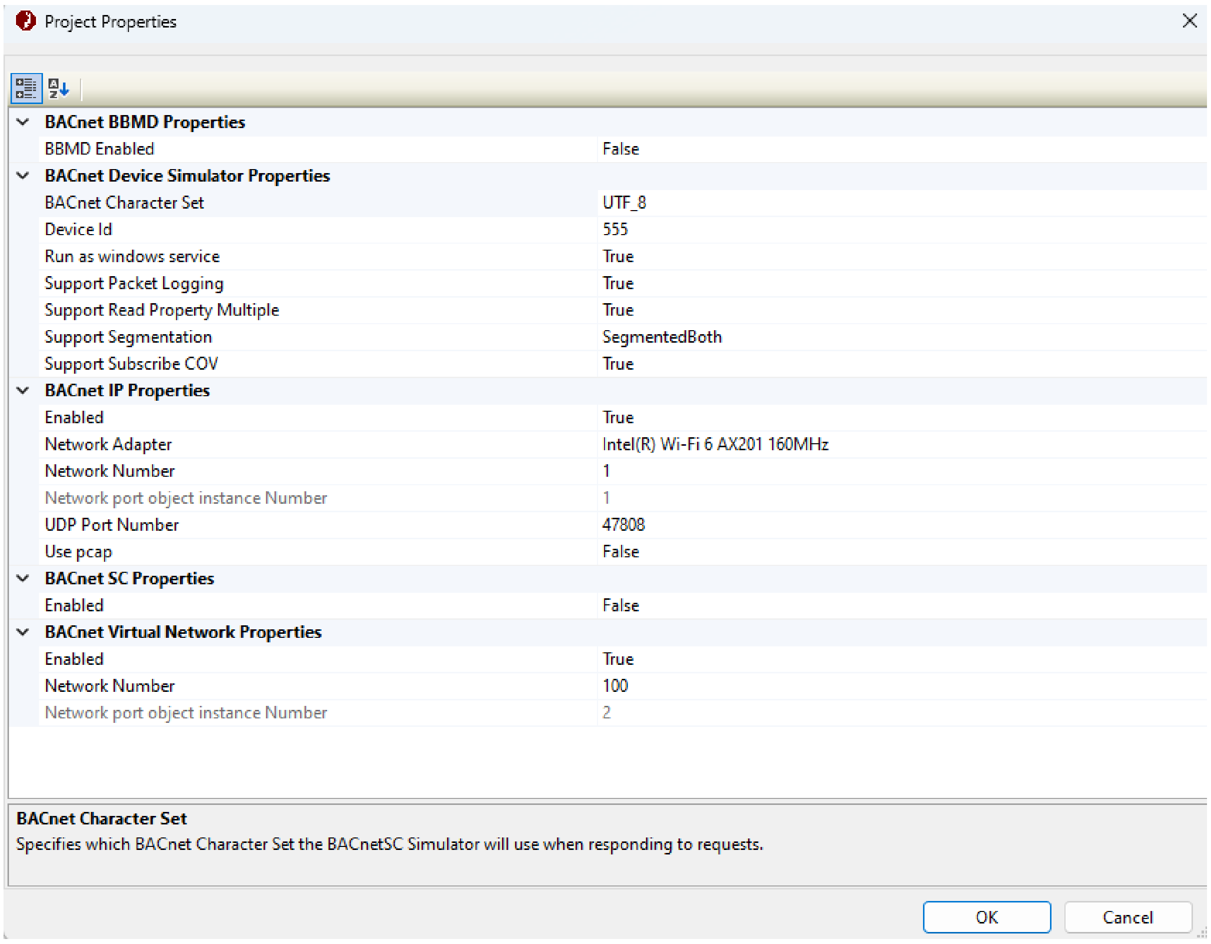
Project properties

Users can set the project properties using the ***Project -> Properties*** menu.



Name	Description
Web Server	Web server plugin
Virtual Network 100	BACnet Virtual Network
Plugins	Gateway Administration
(555) Testing	BACnet local device

When you click on the properties menu, the following dialog will be displayed.



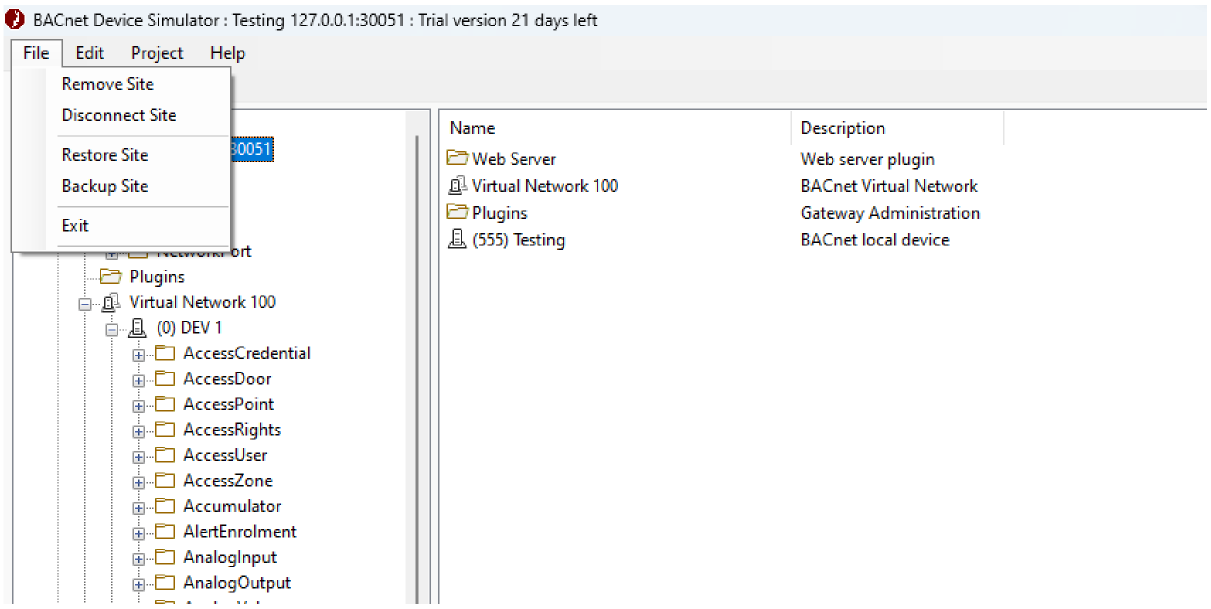
BACnet BBMD Properties	
BBMD Enabled	False
BACnet Device Simulator Properties	
BACnet Character Set	UTF_8
Device Id	555
Run as windows service	True
Support Packet Logging	True
Support Read Property Multiple	True
Support Segmentation	SegmentedBoth
Support Subscribe COV	True
BACnet IP Properties	
Enabled	True
Network Adapter	Intel(R) Wi-Fi 6 AX201 160MHz
Network Number	1
Network port object instance Number	1
UDP Port Number	47808
Use pcap	False
BACnet SC Properties	
Enabled	False
BACnet Virtual Network Properties	
Enabled	True
Network Number	100
Network port object instance Number	2

BACnet Character Set
Specifies which BACnet Character Set the BACnetSC Simulator will use when responding to requests.

From the properties page you can enable/disable ports and configure the device and simulator settings..

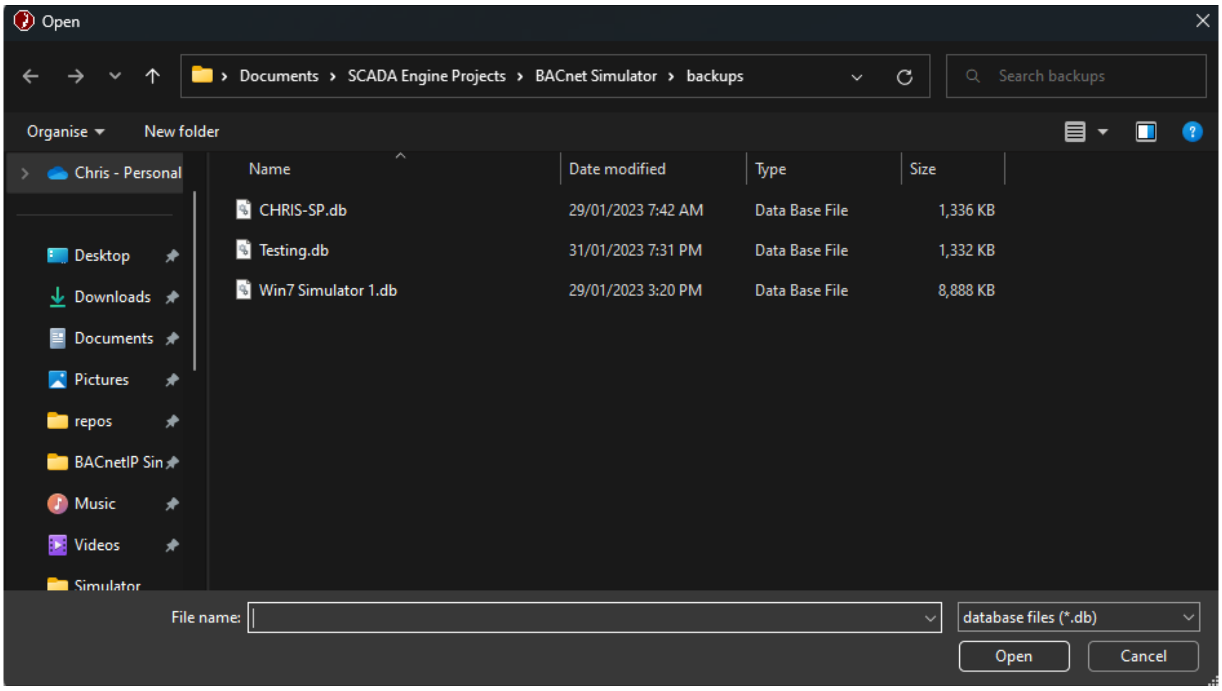
Restore Site

Users can open an already saved .db file selecting a site node and then using the ***File -> Restore Site*** menu.



Name	Description
Web Server	Web server plugin
Virtual Network 100	BACnet Virtual Network
Plugins	Gateway Administration
(555) Testing	BACnet local device

When you clicked on the Open button file open dialog will be displayed and user can select the file that needs to be opened.



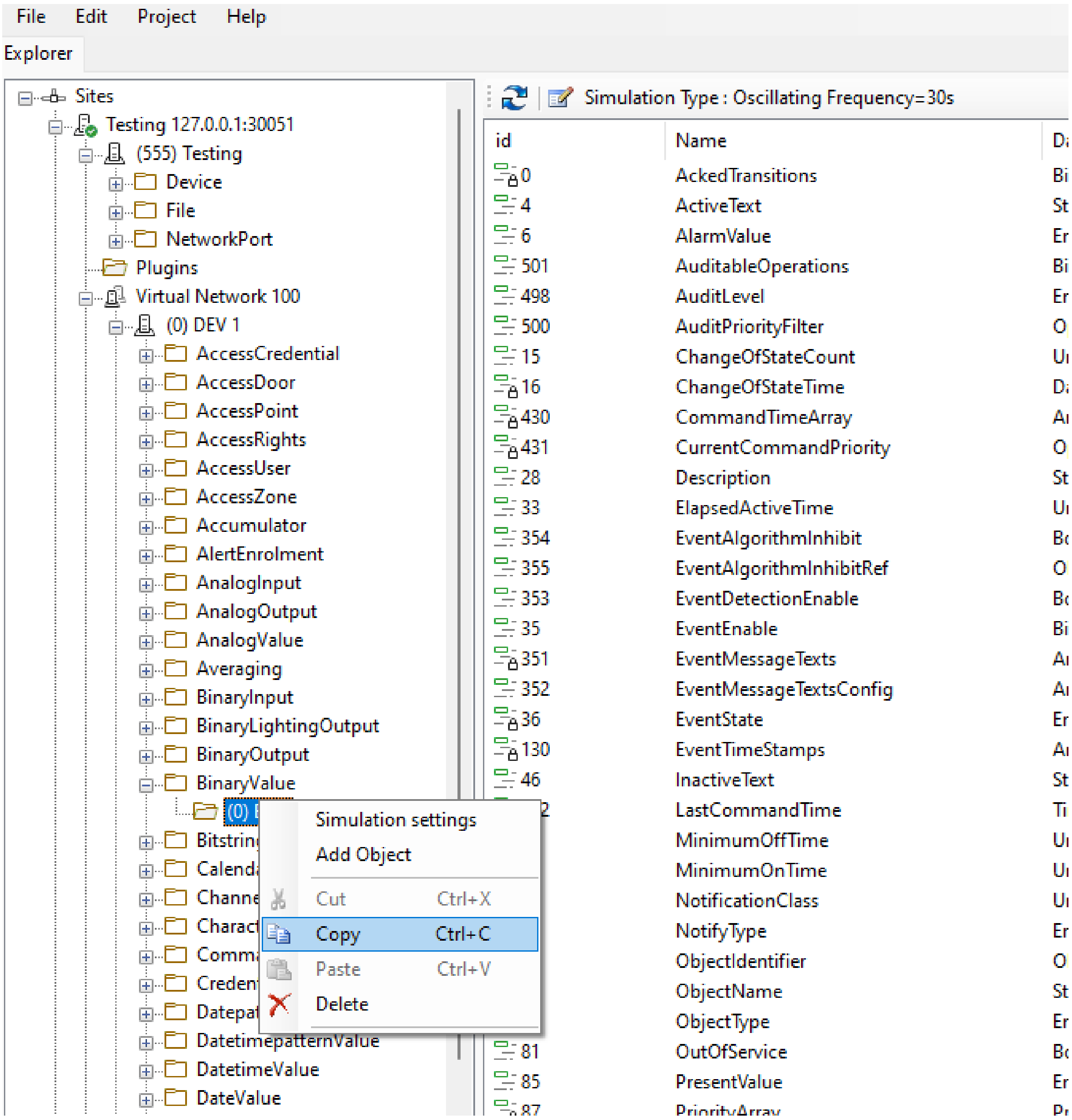
Name	Date modified	Type	Size
CHRIS-SP.db	29/01/2023 7:42 AM	Data Base File	1,336 KB
Testing.db	31/01/2023 7:31 PM	Data Base File	1,332 KB
Win7 Simulator 1.db	29/01/2023 3:20 PM	Data Base File	8,888 KB

When the file is opened, the opened network will be displayed in the BACnet Simulator.

Copy / Paste network objects

Users can copy and paste the network devices, object types or instances in the network.

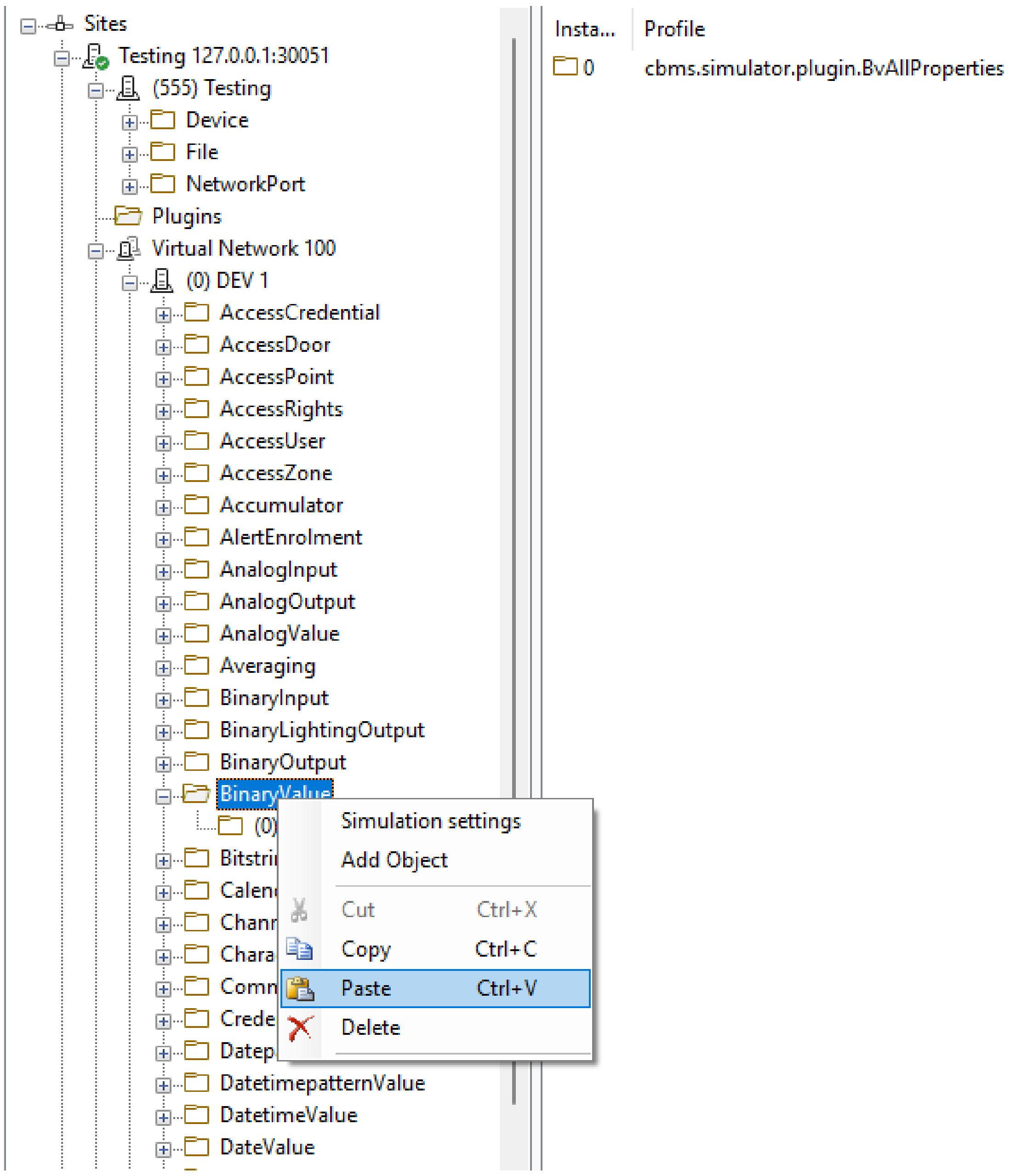
Users can copy the corresponding object by using the ***Edit->Copy*** menu or ***Copy*** menu in the popup menu (Which appears when you right click on the left tree panel).



id	Name	D
0	AckedTransitions	Bi
4	ActiveText	St
6	AlarmValue	Er
501	AuditableOperations	Bi
498	AuditLevel	Er
500	AuditPriorityFilter	O
15	ChangeOfStateCount	U
16	ChangeOfStateTime	D
430	CommandTimeArray	A
431	CurrentCommandPriority	O
28	Description	St
33	ElapsedActiveTime	U
354	EventAlgorithmInhibit	B
355	EventAlgorithmInhibitRef	O
353	EventDetectionEnable	B
35	EventEnable	Bi
351	EventMessageTexts	A
352	EventMessageTextsConfig	A
36	EventState	Er
130	EventTimeStamps	A
46	InactiveText	St
2	LastCommandTime	Ti
	MinimumOffTime	U
	MinimumOnTime	U
	NotificationClass	U
	NotifyType	Er
	ObjectIdentifier	O
	ObjectName	St
	ObjectType	Er
81	OutOfService	B
85	PresentValue	Er
87	PriorityArray	Pr

If user selects at the device level, all the object types and instances under the device will be copied. If user selects at the object type level all the instances under that device will be copied and if it is the instance level, only the instance details will be copied.

After user copying a network object, users can paste it to the network using the ***Edit -> Paste*** menu or ***Paste*** menu in the popup menu.



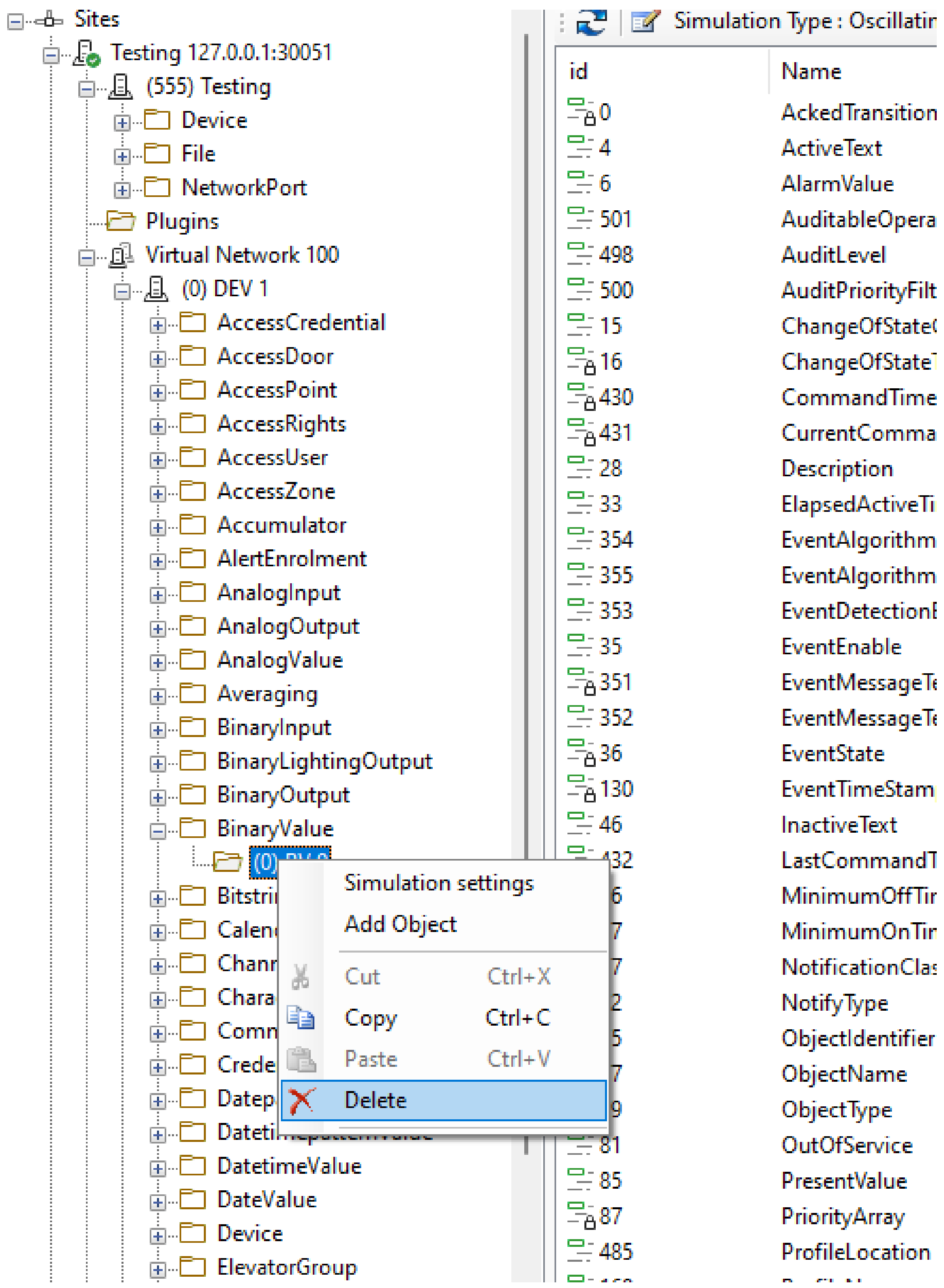
Pasting operation depends on where user needs to paste the copied item.

If user select to paste at the root level (by selecting the BACnet Network node), copied item will be pasted as a new device with the copied content. If user selects a particular device node, object type node or an instance node, copied item will be pasted under the selected device.

In detail, if the copied item is a device, then all the object types and instances are pasted under the selected device, if copied item is an object type or an instance, then all the copied instances or the copied instance will be pasted under the corresponding object type of the selected device.

Delete Network Object

Users can delete network objects using the ***Edit-> Delete*** menu or ***Delete*** menu item in the popup menu.



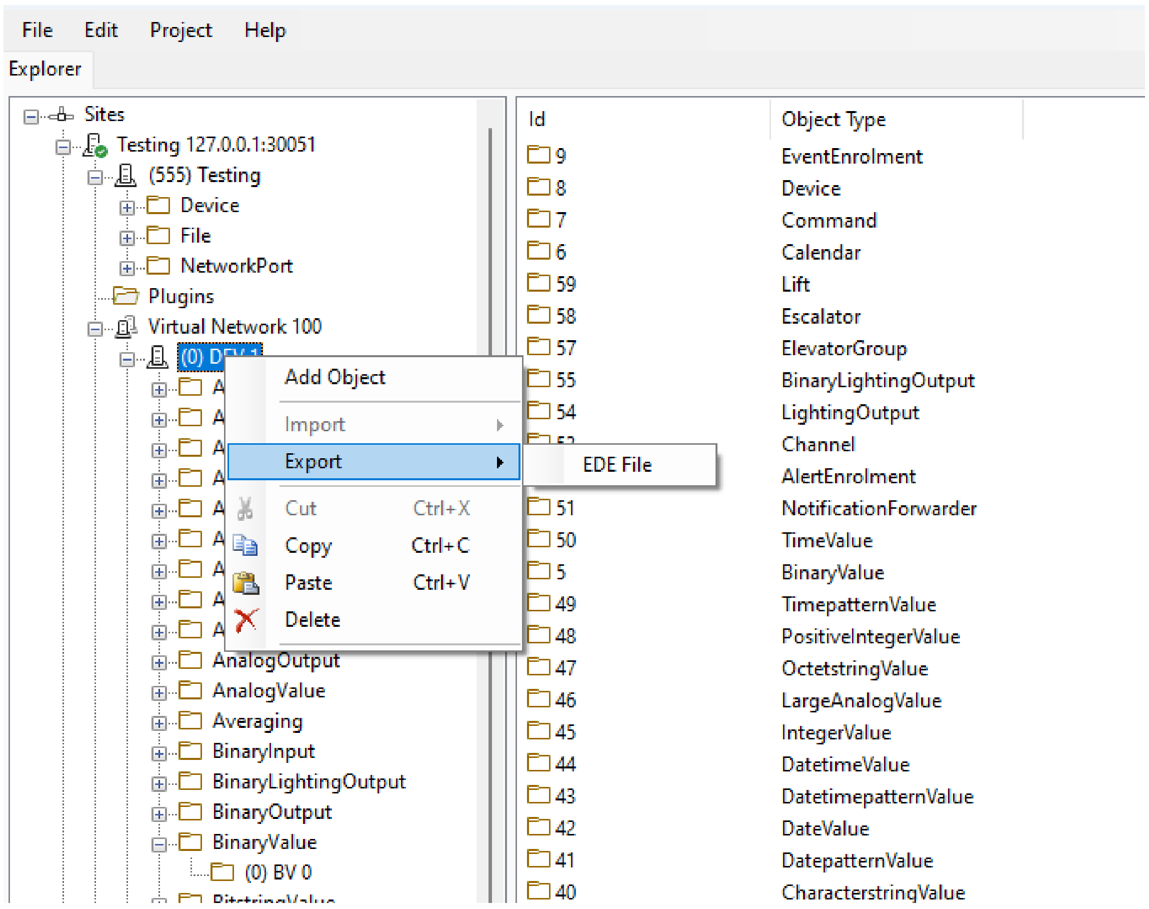
id	Name
0	AckedTransition
4	ActiveText
6	AlarmValue
501	AuditableOpera
498	AuditLevel
500	AuditPriorityFilt
15	ChangeOfState(
16	ChangeOfState
430	CommandTime
431	CurrentComma
28	Description
33	ElapsedActiveTi
354	EventAlgorithm
355	EventAlgorithm
353	EventDetection
35	EventEnable
351	EventMessageTe
352	EventMessageTe
36	EventState
130	EventTimeStam
46	InactiveText
432	LastCommandT
6	MinimumOffTir
7	MinimumOnTir
7	NotificationClas
2	NotifyType
5	ObjectIdentifier
7	ObjectName
9	ObjectType
81	OutOfService
85	PresentValue
87	PriorityArray
485	ProfileLocation

SCADA Engine
11 Buccaneer Avenue,
Lammermoor, Qld 4703 Australia
Tel: (61) 0490 118 346
E-mail: sales@scadaengine.com

If user selects at the device level, all the objects and instances of that device will be deleted. If user selects at object level or instance level only the corresponding object type or the instances will be deleted.

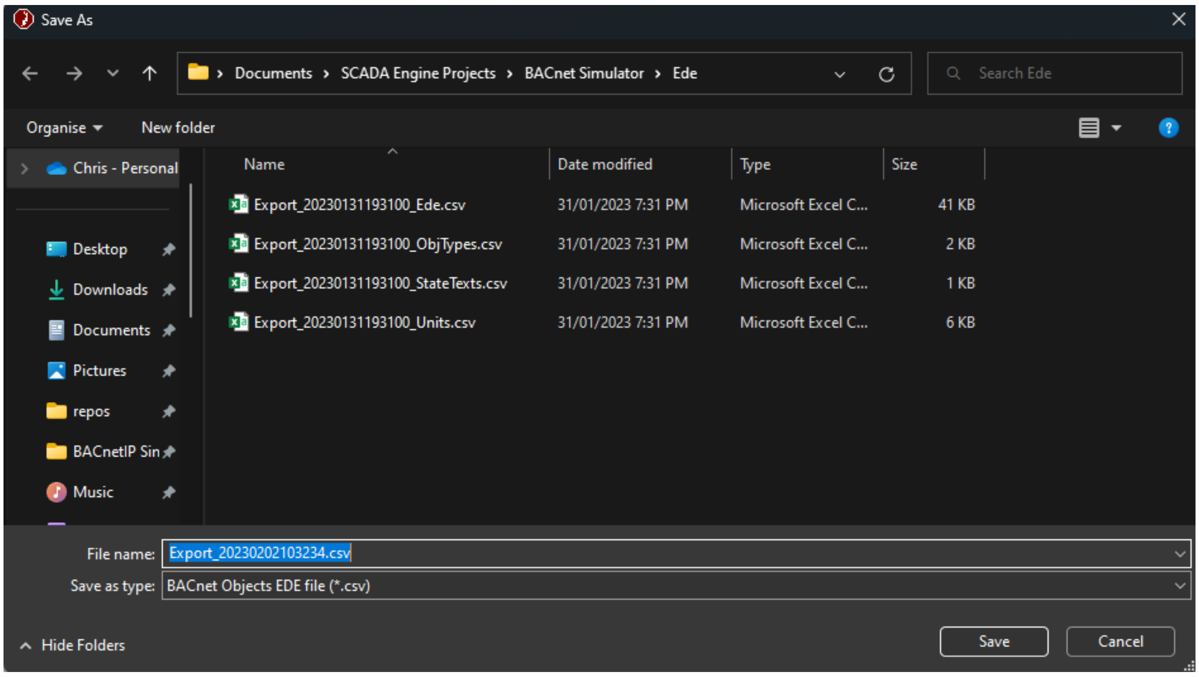
Exporting a Network or Device

Users can export a device or network to an EDE file using the ***File ->Export -> EDE File*** menu.



Id	Object Type
9	EventEnrolment
8	Device
7	Command
6	Calendar
59	Lift
58	Escalator
57	ElevatorGroup
55	BinaryLightingOutput
54	LightingOutput
[illegible]	Channel
[illegible]	AlertEnrolment
51	NotificationForwarder
50	TimeValue
5	BinaryValue
49	TimepatternValue
48	PositiveIntegerValue
47	OctetstringValue
46	LargeAnalogValue
45	IntegerValue
44	DatetimeValue
43	DatetimepatternValue
42	DateValue
41	DatepatternValue
40	CharacterstringValue

When you click on the menu, Save As dialog will be displayed and in that you can specify the file location and a file name for the new file. File will be saved as a .csv file.

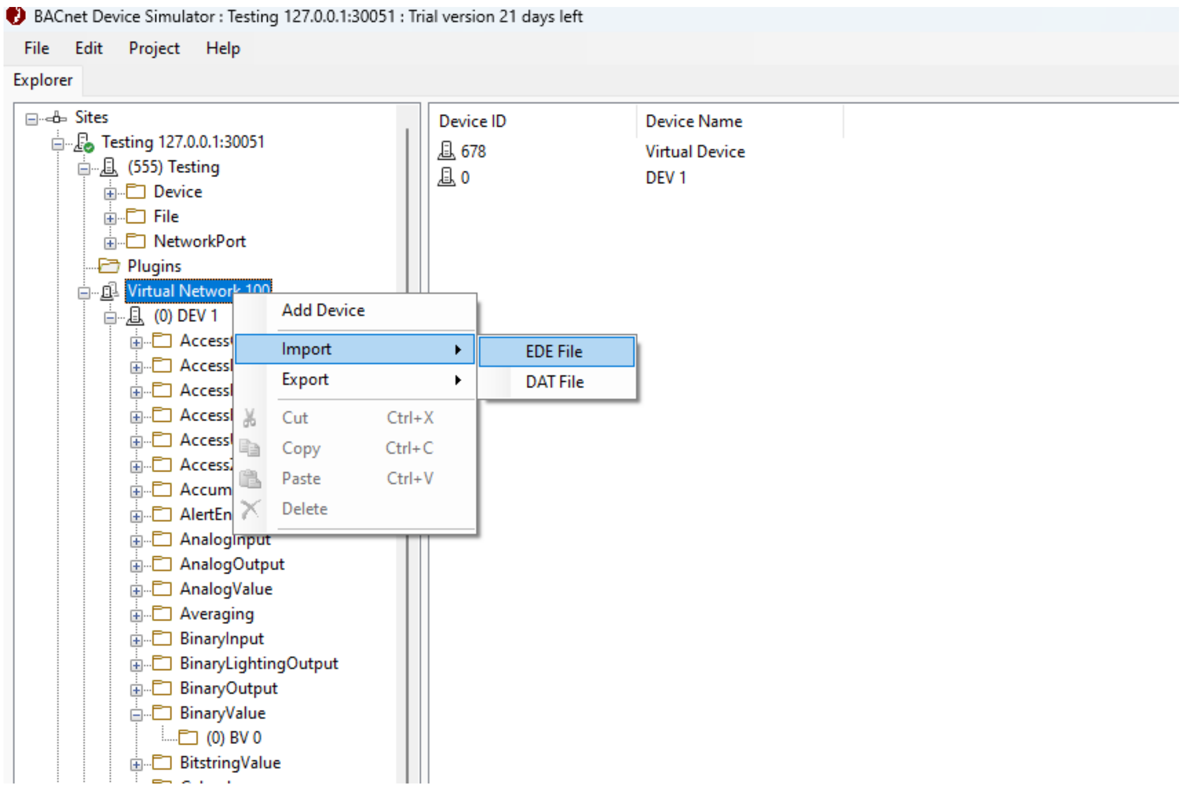


Name	Date modified	Type	Size
Export_20230131193100_Ede.csv	31/01/2023 7:31 PM	Microsoft Excel C...	41 KB
Export_20230131193100_ObjTypes.csv	31/01/2023 7:31 PM	Microsoft Excel C...	2 KB
Export_20230131193100_StateTexts.csv	31/01/2023 7:31 PM	Microsoft Excel C...	1 KB
Export_20230131193100_Units.csv	31/01/2023 7:31 PM	Microsoft Excel C...	6 KB

File name: Export_20230202103234.csv
Save as type: BACnet Objects EDE file (*.csv)

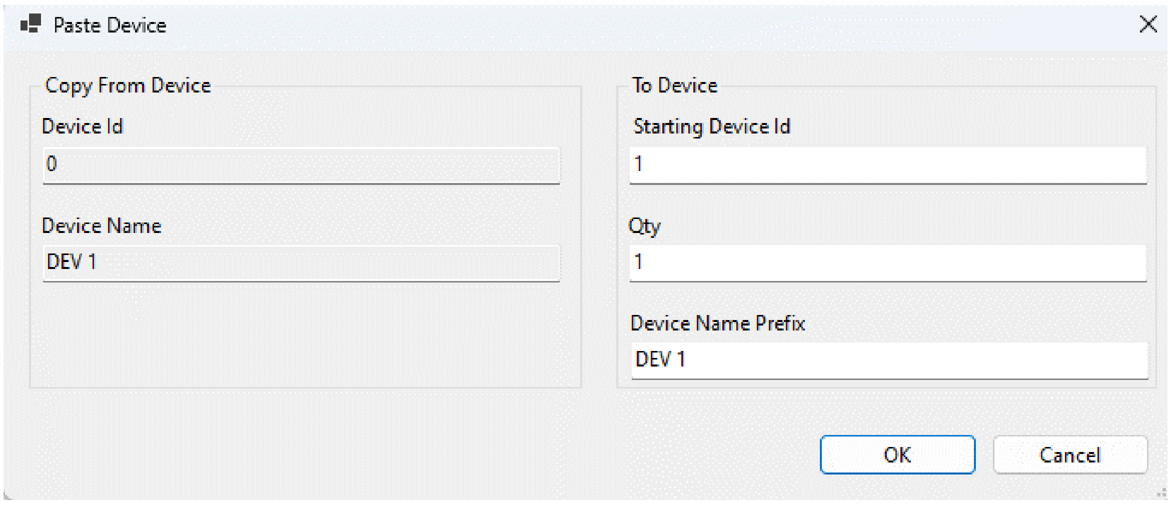
Importing a network or device

You can import an EDE or Dat file using the File->Import->EDE File Menu. The dat file format is compatible with previous versions of the simulator and can be used for importing a configuration from a previous release of the simulator.



Device ID	Device Name
678	Virtual Device
0	DEV 1

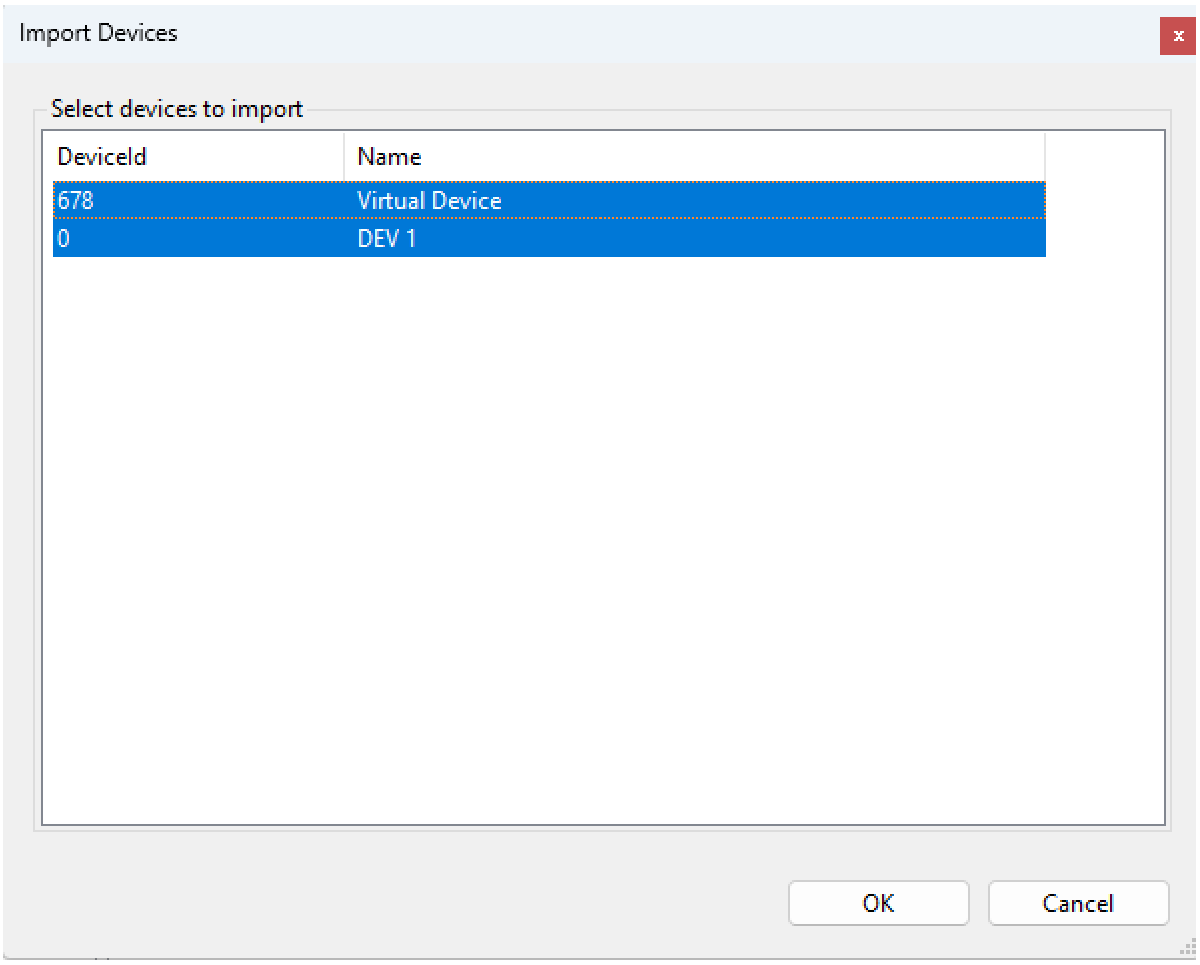
When you select a file that contains a single device then the following dialog appears to allow the device name to be changed and the quantity of devices to be created.



Copy From Device
Device Id: 0
Device Name: DEV 1

To Device
Starting Device Id: 1
Qty: 1
Device Name Prefix: DEV 1

When there are multiple devices in the file then the following dialog box appears and the user can select which devices to import.


Import Devices

Select devices to import

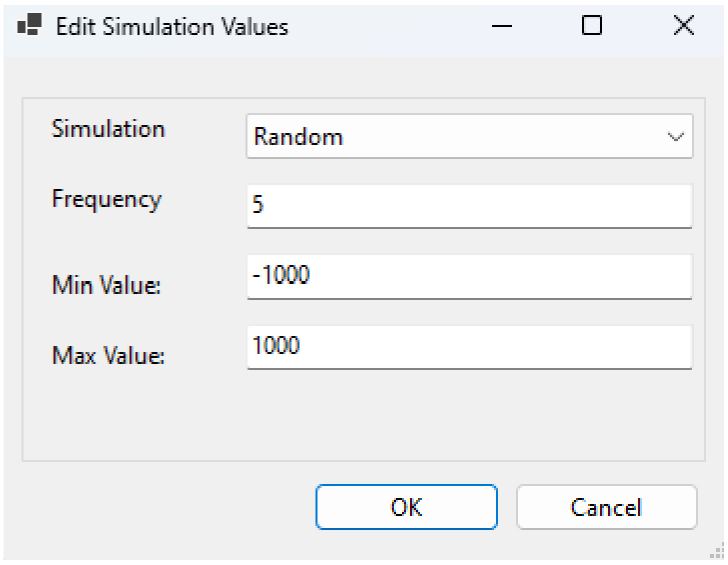
DeviceId	Name
678	Virtual Device
0	DEV 1

Closing the application

When you press the windows close button to exit the information there is no need to save any changes because the database file keeps all changes when they are made. Previous versions of the simulator worked differently and changes were required to be saved bfore exiting the application.

Simulation settings

Analog and binary objects can be configured to simulate a present value that changes on a timed basis. The settings can be changed on an object or globally for a device by right clcking on an an obect node and then selecting the ***Edit ->Simulation Settings*** menu to bring up the following dialog box.



Simulation: Random
Frequency: 5
Min Value: -1000
Max Value: 1000

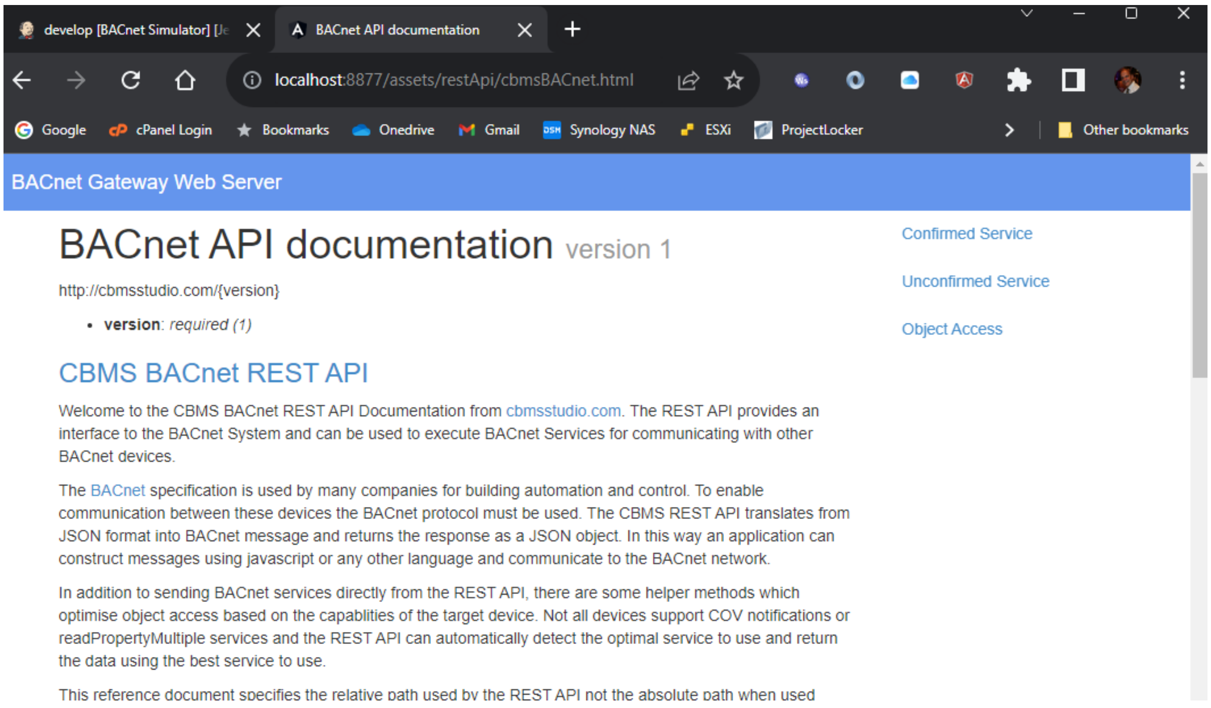
Users can specify the simulation frequency, simulation type and the value interval for the simulation (value interval is needed only for the incremental simulation type).

Available simulation types are Incremental, Random, Sinusoidal and Ramp. Incremental simulation will gradually increase the value of the present value of all the instances by an amount equal to the value interval, in each time interval equal to the simulation frequency.

Random simulation will randomly set the value of the present value of all the instances. In Sinusoidal simulation present value of all the instances will be set according to the Sin function, and in Ramp simulation present value will be set according to the Ramp function.

BACnet REST Web Server

The simulator has an optional web server plugin which is open by default on localhost port 8877. If you have enabled this plugin then you can open a browser to view the web page which looks like the following.



BACnet API documentation version 1

http://cbmsstudio.com/{version}

- **version**: *required (1)*

CBMS BACnet REST API

Welcome to the CBMS BACnet REST API Documentation from cbmsstudio.com. The REST API provides an interface to the BACnet System and can be used to execute BACnet Services for communicating with other BACnet devices.

The BACnet specification is used by many companies for building automation and control. To enable communication between these devices the BACnet protocol must be used. The CBMS REST API translates from JSON format into BACnet message and returns the response as a JSON object. In this way an application can construct messages using javascript or any other language and communicate to the BACnet network.

In addition to sending BACnet services directly from the REST API, there are some helper methods which optimise object access based on the capablities of the target device. Not all devices support COV notifications or readPropertyMultiple services and the REST API can automatically detect the optimal service to use and return the data using the best service to use.

This reference document specifies the relative path used by the REST API not the absolute path when used

The REST API provides an interface to the BACnet System and can be used to execute BACnet Services for communicating with other BACnet devices.

The BACnet specification is used by many companies for building automation and control. To enable communication between these devices the BACnet protocol must be used. The CBMS REST API translates from JSON format into BACnet message and returns the response as a JSON object. In this way an application can construct messages using javascript or any other language and communicate to the BACnet network.

In addition to sending BACnet services directly from the REST API, there are some helper methods which optimise object access based on the capablities of the target device. Not all devices support COV notifications or readPropertyMultiple services and the REST API can automatically detect the optimal service to use and return the data using the best service to use.

This reference document specifies the relative path used by the REST API not the absolute path when used inside a web server. For example a web server may choose to use a web service path begining with /ws/bacnet. In this case, if the network method is required the path would be /ws/bacnet/network

Confirmed Service

Confirmed BACnet services can be called directly from the REST API by constructing a message in JSON and posting it to the REST API. The API will convert the JSON text into a BACnet message and send it to the BACnet device using the BACnet protocol. When a confirmed message is sent the REST API will wait until the device responds or a timeout occurs before sending the response back in JSON format. The following paths can be used for sending BACnet Services.

- /confirmed/service/acknowledgeAlarm POST
- /confirmed/service/getAlarmSummary POST
- /confirmed/service/getEnrolmentSummary POST
- /confirmed/service/subscribeCov POST
- /confirmed/service/atomicReadFile POST
- /confirmed/service/atomicWriteFile POST
- /confirmed/service/addListElement POST
- /confirmed/service/removeListElement POST
- /confirmed/service/createObject POST
- /confirmed/service/deleteObject POST
- /confirmed/service/readProperty POST
- /confirmed/service/readPropertyMultiple POST
- /confirmed/service/writeProperty POST
- /confirmed/service/writePropertyMultiple POST
- /confirmed/service/deviceCommunicationControl POST
- /confirmed/service/confirmedPrivateTransfer POST
- /confirmed/service/confirmedTextMessage POST
- /confirmed/service/reinitializeDevice POST
- /confirmed/service/readRange POST
- /confirmed/service/lifeSafetyOperation POST
- /confirmed/service/subscribeCovProperty POST
- /confirmed/service/getEventInformation POST
- /confirmed/service/subscribeCovPropertyMultiple POST

Unconfirmed Service

Unconfirmed BACnet services can be called directly from the REST API by constructing a message in JSON and posting it to the REST API. The API will convert the JSON text into a BACnet message and send it to the BACnet device using the BACnet protocol. When an unconfirmed message is sent, the

REST API will return immediately without waiting for a response. The following paths can be used for sending BACnet Services.

- /unconfirmed/service/privateTransfer POST
- /unconfirmed/service/textMessage POST
- /unconfirmed/service/timeSynchronization POST
- /unconfirmed/service/whoHas POST
- /unconfirmed/service/whoIs POST
- /unconfirmed/service/utcTimeSynchronization POST
- /unconfirmed/service/writeGroup POST

Object Access

The Object Access helper methods provide a simplified approach to accessing BACnet objects and properties from any device on the network. The REST API will determine the appropriate BACnet Service to use when accessing the data. This could be readProperty, writeProperty, readPropertyMultiple, writePropertyMultiple or COV.

- /objectAccess/network GET
- /objectAccess/read POST
- /objectAccess/write POST
- /objectAccess/{deviceId}
- /objectAccess/{deviceId}/objectTypes GET
- /objectAccess/{deviceId}/objectList GET
- /objectAccess/{deviceId}/read POST
- /objectAccess/{deviceId}/write POST
- /objectAccess/{deviceId}/{objectType}
- /objectAccess/{deviceId}/{objectType}/instances GET
- /objectAccess/{deviceId}/{objectType}/objectList GET
- /objectAccess/{deviceId}/{objectType}/read POST
- /objectAccess/{deviceId}/{objectType}/write POST
- /objectAccess/{deviceId}/{objectType}/{instance} GET
- /objectAccess/{deviceId}/{objectType}/{instance}/read POST
- /objectAccess/{deviceId}/{objectType}/{instance}/write POST
- /objectAccess/{deviceId}/{objectType}/{instance}/{propertyIdentifier} GET PUT

Customisation

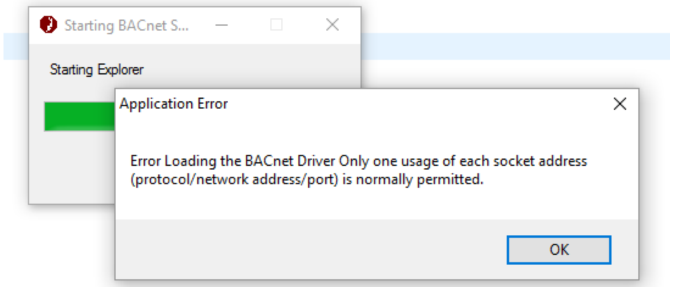
The default behaviour of the simulator can be customised by writing a plugin and placing it in the plugin directory. When the simulator starts the plugin will be loaded and it can be used to update the properties within the simulator. Each object type can be customised with a class that models the behaviour of the object type.

For example, if a class is defined for an AnalogInput object, then for every AnalogInput object within the simulator a new instance of this class will be created. The simulator will call virtual members on this class to indicate when the object is loaded, started, stopped, etc. Using these virtual members the plugin can define custom behaviour to take.

A separate document provides the detailed information for creating a plugin in csharp using visual studio.

Troubleshooting

When you are starting the BACnet Simulator you might be displayed the following error message.



Error Loading the BACnet Driver Only one usage of each socket address (protocol/network address/port) is normally permitted.

This error message indicates that there are some other BACnet application is running in the machine at that time.

To resolve this issue you need to shut down the other application and open the BACnet Simulator again.

Protocol Implementation Conformance Statement

Products

Product	Model Number	Protocol Revision	Software Version	Firmware Version
BACnet Simulator	SE-SIM	135-2020 (22)	6.0.55	6.0.55

Date Tested: 2 January 2023

Vendor Information

SCADA Engine
11 Buccaneer Avenue
Lammemoor Queensland

www.scadaenginecom

Product Description

The SCADA Engine BACnet Simulator is used to simulate 1 or more BACnet Devices.

BACnet Standardized Device Profile

Product	Device Profile	Tested
SE-SIM	BACnet Application Specific Controller (B-ASC)	

Supported BIBBs

Product	Supported BIBBs	BIBB Name	
SE-SIM	DS-RP-A	Data Sharing-ReadProperty-A	
	DS-RP-B	Data Sharing-ReadProperty-B	
	DS-RPM-A	Data Sharing-ReadPropertyMultiple-A	
	DS-RPM-B	Data Sharing-ReadPropertyMultiple-B	
	DS-WP-A	Data Sharing-WriteProperty-A	
	DS-WP-B	Data Sharing-WriteProperty-B	
	DS-WPM-A	Data Sharing-WritePropertyMultiple-A	
	DS-WPM-B	Data Sharing-WritePropertyMultiple-B	
	DS-COV-A	Data Sharing-COV-A	
	DS-COV-B	Data Sharing-COV-B	
	DS-COVP-A	Data Sharing-COVP-A	
	DS-COVP-B	Data Sharing-COVP-B	
	DS-COVU-A	Data Sharing-COV-Unsolicited-A	
	DS-COVU-B	Data Sharing-COV-Unsolicited-B	
	SCHED-A	Scheduling-A	
	SCHED-I-B	Scheduling-Internal-B	
	SCHED-E-B	Scheduling-External-A	

Product	Supported BIBBs	BIBB Name	
	T-VMT-A	Trending-Viewing and Modifying Trends-A	
	T-VMT-I-B	Trending-Viewing and Modifying Trends-Internal-B	
	T-VMT-E-B	Trending-Viewing and Modifying Trends-External-B	
	T-ATR-A	Trending-Automated Trend Retrieval-A	
	T-ATR-B	Trending-Automated Trend Retrieval-B	
	NM-CE-A	Network Management-Connection Establishment-A	
	NM-CE-B	Network Management-Connection Establishment-B	
	AE-N-A	Alarm and Event-Notification-A	
	AE-N-I-B	Alarm and Event-Notification Internal-B	
	AE-N-E-B	Alarm and Event-Notification External-B	
	AE-ACK-A	Alarm and Event-ACK-A	
	AE-ACK-B	Alarm and Event-ACK-B	
	AE-ASUM-A	Alarm and Event-Alarm Summary-A	
	AE-ASUM-B	Alarm and Event-Alarm Summary-B	
	AE-ESUM-A	Alarm and Event-Enrollment Summary-A	
	AE-ESUM-B	Alarm and Event-Enrollment Summary-B	
	AE-INFO-A	Alarm and Event-Information-A	
	AE-INFO-B	Alarm and Event-Information-B	
	AE-LS-A	Alarm and Event-LifeSafety-A	
	AE-LS-B	Alarm and Event-LifeSafety-B	

Product	Supported BIBBs	BIBB Name	
	DM-RD-A	Device Management-ReinitializeDevice-A	
	DM-RD-B	Device Management-ReinitializeDevice-B	
	DM-DDB-A	Device Management-Dynamic Device Binding-A	
	DM-DDB-B	Device Management-Dynamic Device Binding-B	
	DM-DOB-A	Device Management-Dynamic Object Binding-A	
	DM-DOB-B	Device Management-Dynamic Object Binding-B	
	DM-DCC-A	Device Management-DeviceCommunicationControl-A	
	DM-DCC-B	Device Management-DeviceCommunicationControl-B	
	DM-PT-A	Device Management-Private Transfer-A	
	DM-PT-B	Device Management-Private Transfer-B	
	DM-TM-A	Device Management-Text Message-A	
	DM-TM-B	Device Management-Text Message-B	
	DM-TS-A	Device Management-TimeSynchronization-A	
	DM-TS-B	Device Management-TimeSynchronization-B	
	DM-UTC-A	Device Management-UTCTimeSynchronization-A	
	DM-UTC-B	Device Management-UTCTimeSynchronization-B	
	DM-LM-A	Device Management-List Manipulation-A	
	DM-LM-B	Device Management-List Manipulation-B	
	DM-OCD-A	Device Management-Object Creation and Deletion-A	
	DM-OCD-B	Device Management- Object Creation and Deletion -B	

Standard Object Types Supported

Product	Object Type	Creatable	Deletable	Tested
SE-SIM	Analog Input	Yes	Yes	
	Analog Output	Yes	Yes	
	Analog Value	Yes	Yes	
	Binary Input	Yes	Yes	
	Binary Output	Yes	Yes	
	Binary Value	Yes	Yes	
	Calendar	Yes	Yes	
	Device	No	No	
	Event Enrollment	Yes	Yes	
	File	Yes	Yes	
	Loop	Yes	Yes	
	Multi-state Value	Yes	Yes	
	Notification Class	Yes	Yes	
	Program	Yes	Yes	
	Schedule	Yes	Yes	
	Trend Log	Yes	Yes	
	LifeSafetyPoint	Yes	Yes	
	LifeSafetyZone	Yes	Yes	
	Accumulator	Yes	Yes	
	PulseConverter	Yes	Yes	

Data Link Layer Options

Product	Data Link	Options	Tested
SE-SIM	BACnet/IP (Annex J)	Can communicate as a Direct BACnet/IP device. Can register as a Foreign BACnet/IP device.	

Segmentation Capability

Product	Segmentation Type	Supported	Window Size (MS/TP product limited to 1)	Tested
SE-SIM	Able to transmit segmented messages	Yes	Configurable	
	Able to receive segmented messages	Yes	Configurable	

Device Address Binding

Product	Static Binding Supported	Tested
SE-SIM	Yes	

Networking Options

Product	Router Option	Options	Tested
SE-SIM			

Character Sets

Product	Character Sets supported	
SE-SIM	ANSI X3.4	
	IBM Microsoft DBCS	
	ISO 8859-1	