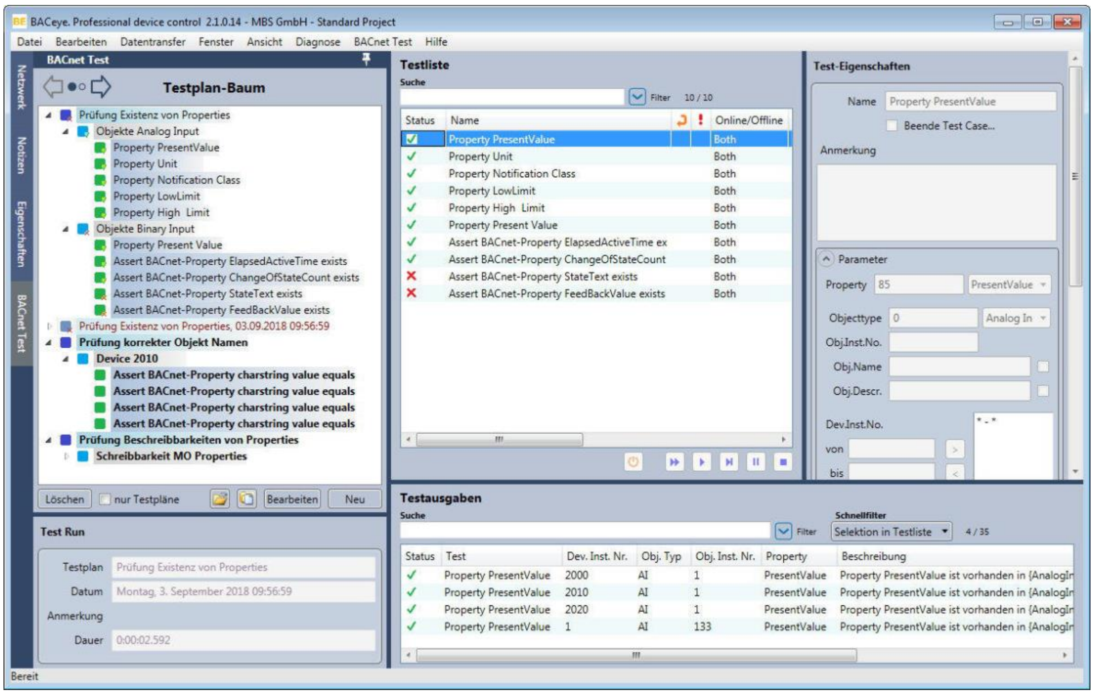
MBS®
Made by Specialists.



BACnet Test

Testplan-Baum

- Prüfung Existenz von Properties
  - Objekte Analog Input
    - Property PresentValue
    - Property Unit
    - Property Notification Class
    - Property LowLimit
    - Property High Limit
  - Objekte Binary Input
    - Property Present Value
    - Assert BACnet-Property ElapsedActiveTime exists
    - Assert BACnet-Property ChangeOfStateCount exists
    - Assert BACnet-Property StateText exists
    - Assert BACnet-Property FeedBackValue exists
- Prüfung Existenz von Properties, 03.09.2018 09:56:59
- **Prüfung korrekter Objekt Namen**
  - **Device 2010**
    - **Assert BACnet-Property charstring value equals**
    - **Assert BACnet-Property charstring value equals**
    - **Assert BACnet-Property charstring value equals**
    - **Assert BACnet-Property charstring value equals**
- **Prüfung Beschreibbarkeiten von Properties**
  - **Schreibbarkeit MO Properties**

Test Run

| | |
|---|---|
| Testplan | Prüfung Existenz von Properties |
| Datum | Montag, 3. September 2018 09:56:59 |
| Anmerkung | |
| Dauer | 0:00:02.592 |

Testliste

Suche

Filter 10 / 10

| Status | Name | | ! | Online/Offline |
|---|---|---|---|---|
| ✓ | Property PresentValue | | | Both |
| ✓ | Property Unit | | | Both |
| ✓ | Property Notification Class | | | Both |
| ✓ | Property LowLimit | | | Both |
| ✓ | Property High Limit | | | Both |
| ✓ | Property Present Value | | | Both |
| ✓ | Assert BACnet-Property ElapsedActiveTime ex | | | Both |
| ✓ | Assert BACnet-Property ChangeOfStateCount | | | Both |
| ✗ | Assert BACnet-Property StateText exists | | | Both |
| ✗ | Assert BACnet-Property FeedBackValue exists | | | Both |

Test-Eigenschaften

| | |
|---|---|
| Name | Property PresentValue |
| Beende Test Case... | ☐ |
| Anmerkung | |

Parameter

| | | |
|---|---|---|
| Property | 85 | PresentValue |
| Objecttype | 0 | Analog In |
| Obj.Inst.No. | | |
| Obj.Name | | ☐ |
| Obj.Descr. | | ☐ |
| Dev.Inst.No. | | * - * |
| von | | |
| bis | | |

Testausgaben

Suche

Filter

Schnellfilter: Selektion in Testliste 4 / 35

| Status | Test | Dev. Inst. Nr. | Obj. Typ | Obj. Inst. Nr. | Property | Beschreibung |
|---|---|---|---|---|---|---|
| ✓ | Property PresentValue | 2000 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogIr |
| ✓ | Property PresentValue | 2010 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogIr |
| ✓ | Property PresentValue | 2020 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogIr |
| ✓ | Property PresentValue | 1 | AI | 133 | PresentValue | Property PresentValue ist vorhanden in {AnalogIr |

# BACeye BACnet-Test plug-in

**Software version: 1.0.0.7**
**Doc. version: 11/09/2018**

# Contents

**1 Foreword ... 3**
1.1 Copyright ... 3
1.2 Contact information ... 3
1.3 Aim and role of the BACnet-Test plug-in ... 3
**2 Install the test plug-in ... 5**
**3 New test plug-in elements in the BACeye interface ... 6**
3.1 New tab, menus and context menus ... 6
3.2 Components ... 8
3.2.1 Test plan ... 8
3.2.2 Test case ... 9
3.2.3 Assertion (test) ... 9
**4 An example as an introduction ... 10**
4.1 Workflow ... 10
4.2 BACnet devices in the network (example of online test) ... 10
4.3 Use the test plan editor ... 10
4.3.1 Create a new test plan ... 10
4.3.2 Create a new test case ... 11
4.3.3 Create and configure a new test ... 12
4.4 Execute a test run ... 15
4.4.1 Start the test run ... 16
4.4.2 Display test results ... 17
4.5 Execute a test run offline ... 18
4.6 Examples of other, more complex test plans ... 19
**5 Data storage and transfer ... 22**
5.1 Save and delete test plans and test runs ... 22
5.2 Save and load the BACeye project ... 22
5.3 Export and import test plans, test runs and test cases ... 22

# 1 Foreword

Thank you for using the BACeye BACnet-Test plug-in to test BACnet networks.

This short document uses an example to provide an introduction to the BACnet-Test plug-in for BACeye software.

## 1.1 Copyright

## 1.2 Contact information

MBS GmbH
Römerstraße 15
47809 Krefeld, Germany

Telephone: +49 21 51 72 94 – 0
info@mbs-solutions.de
http://www.baceye.com

**Support**
Tel.: +49 21 51 72 94 – 0
support-baceye@mbs-solutions.de

## 1.3 Aim and role of the BACnet-Test plug-in

**BACeye**

BACeye allows easy access to BACnet devices and to their objects and properties. Typical BACeye applications include commissioning BACnet installations, analysing observed problems and monitoring and operating BACnet devices.

**BACeye BACnet-Test plug-in**

*Aim and role:*

The aim of the BACnet-Test plug-in is to eliminate the source of errors caused by the *media discontinuity* at the point separating planning and implementation. The plug-in provides the ability to automatically test for compliance with written requirements (Requirements specification) for a BACnet network. Entire systems or just individual devices or objects can be tested for a required status.

*Background:*

During the process of creating a BACnet network, there are various *transitions between media* which often cause information to become lost or inaccurate. These media transitions occur between planning/defining the contract specification, the invitation to tender, the issued order and the implementation. For example, a planner (client, builder, end customer) plans a BACnet network with systems to automate the building and passes the documents to the project developer.  The project developer receives the instructions and plans the BACnet network (e.g. using a planning system). At this point of separation between planning and implementation, there is a media discontinuity with certain risks of error.

Final acceptance by the planner usually involves testing the functionality and that it fully complies with the rules provided. When the builder has handed the property over, it is not unusual for the planner to find that it does not fulfil the required function, either partially or in its entirety.

Instead of laborious manual tests, the planner that gives the order, or the project manager that accepts the order, can run automated, reliable tests for the entire network, or parts of it, before acceptance (Workflow see Ch. 4.1).

The tests are configured and saved in *test plans* and *test cases*. Should there be any errors, a detailed list of errors is generated. Errors are then corrected and the BACnet network or parts of it are re-tested. Multiple test correction iterations result in the BACnet network achieving its optimal condition, and it can then go live with no problems.

The following individual BACnet elements can be tested for various characteristics:

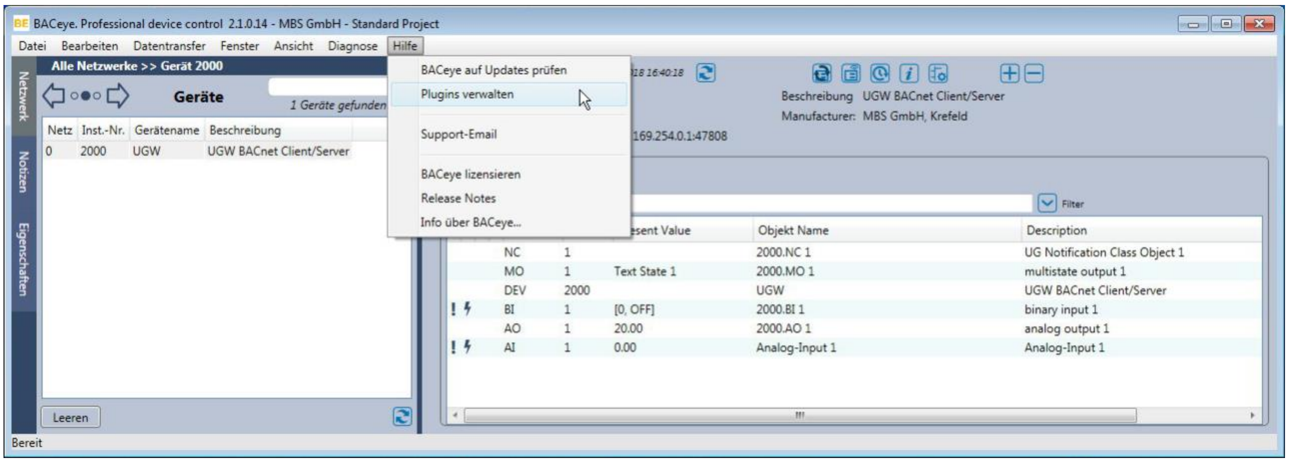
- Network
- Device
- Object
- Property

Whether the test result is to be assessed as positive or negative in each case depends on the relevant test case. The presence of a particular object may be assessed as positive in one case but as negative in another.

# 2 Install the test plug-in

You install the BACnet-Test plug-in as follows:

- Start the BACeye software.
- Click the ***Manage plug-ins*** (Plug-Ins verwalten) item in the ***Help*** (Hilfe) menu (Figure 1).
  The *BACeye – Manage plug-ins* (BACeye – Plug-Ins verwalten) dialog opens (Figure 2).



| Netz | Inst.-Nr. | Gerätename | Beschreibung |
|---|---|---|---|
| 0 | 2000 | UGW | UGW BACnet Client/Server |

18 16:40:18

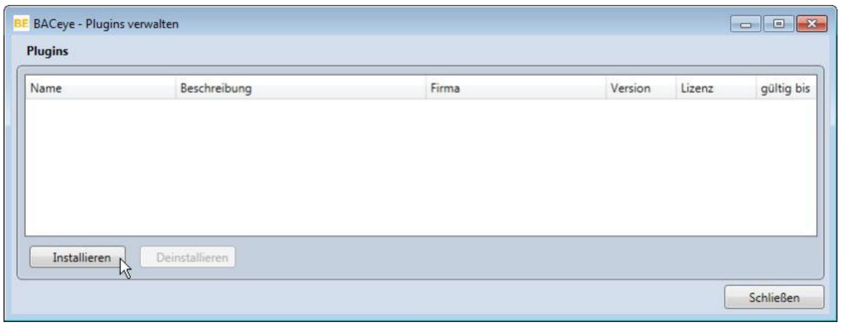
Beschreibung UGW BACnet Client/Server
Manufacturer: MBS GmbH, Krefeld

169.254.0.1:47808

| | | | esent Value | Objekt Name | Description |
|---|---|---|---|---|---|
| | NC | 1 | | 2000.NC 1 | UG Notification Class Object 1 |
| | MO | 1 | Text State 1 | 2000.MO 1 | multistate output 1 |
| | DEV | 2000 | | UGW | UGW BACnet Client/Server |
| ! ϟ | BI | 1 | [0, OFF] | 2000.BI 1 | binary input 1 |
| | AO | 1 | 20.00 | 2000.AO 1 | analog output 1 |
| ! ϟ | AI | 1 | 0.00 | Analog-Input 1 | Analog-Input 1 |

Figure 1

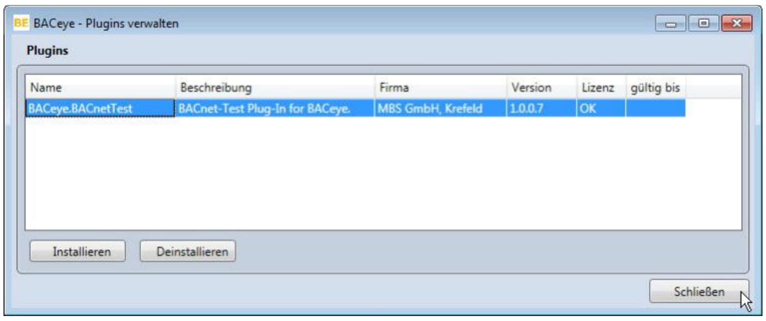
- Click the **[Install]** (Installieren) button in the *BACeye – Manage plug-ins* (BACeye – Plug-Ins verwalten) dialog.
  The selection dialog opens.



| Name | Beschreibung | Firma | Version | Lizenz | gültig bis |
|---|---|---|---|---|---|

Figure 2

- In the selection dialog, select the zip installation file that has been supplied for the BACnet-Test plug-in, and confirm by clicking **[Open]** (Öffnen).
  The plug-in is then installed and appears in the *BACeye – Manage plug-ins* (BACeye – Plug-Ins verwalten) dialog (Figure 3).
- Close the dialog with **[Close]** (Schließen).
  A text window then prompts you to restart the BACeye software.


BACeye - Plugins verwalten

Plugins

| Name | Beschreibung | Firma | Version | Lizenz | gültig bis |
|---|---|---|---|---|---|
| BACeye.BACnetTest | BACnet-Test Plug-In for BACeye. | MBS GmbH, Krefeld | 1.0.0.7 | OK | |

Figure 3

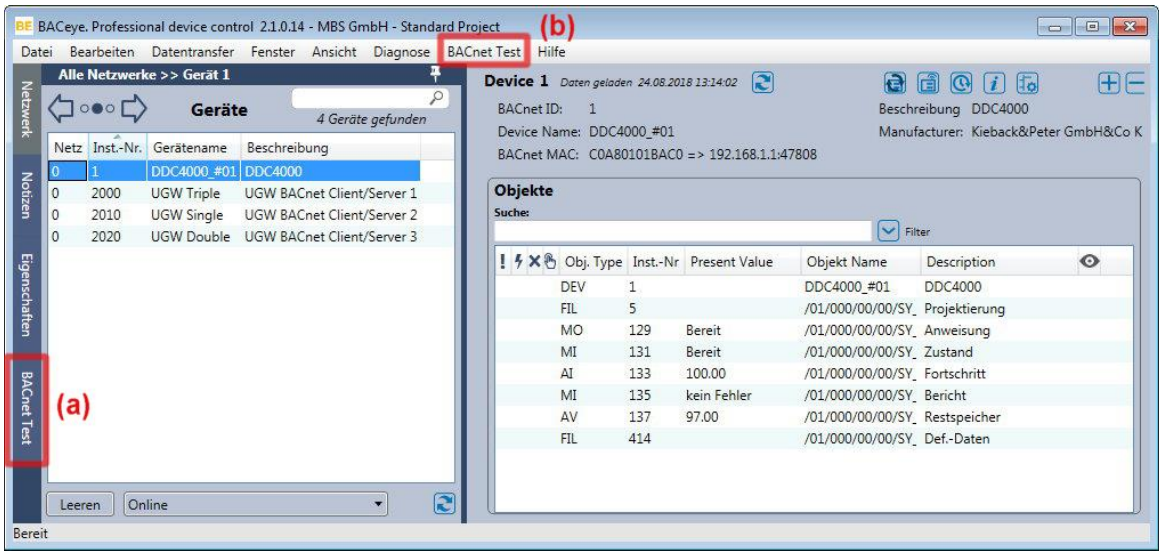
Restart BACeye.
You can then use the BACeye-Test plug-in functions.

# 3 New test plug-in elements in the BACeye interface

## 3.1 New tab, menus and context menus

As a result of the test plug-in, the BACeye software graphic user interface (GUI) now has these new elements (Figure 4 to Figure 6):

- ***BACnet test*** tab in the navigation bar (a)
- ***BACnet test*** menu in the BACeye menu bar (b)
- New functions in the context menus for functional elements


BACeye. Professional device control 2.1.0.14 - MBS GmbH - Standard Project

| Netz | Inst.-Nr. | Gerätename | Beschreibung |
|---|---|---|---|
| 0 | 1 | DDC4000_#01 | DDC4000 |
| 0 | 2000 | UGW Triple | UGW BACnet Client/Server 1 |
| 0 | 2010 | UGW Single | UGW BACnet Client/Server 2 |
| 0 | 2020 | UGW Double | UGW BACnet Client/Server 3 |

Device 1 *Daten geladen 24.08.2018 13:14:02*

BACnet ID: 1
Device Name: DDC4000_#01
BACnet MAC: C0A80101BAC0 => 192.168.1.1:47808
Beschreibung DDC4000
Manufacturer: Kieback&Peter GmbH&Co K

Objekte
Suche:

| ! | ϟ | ✕ | | Obj. Type | Inst.-Nr | Present Value | Objekt Name | Description |
|---|---|---|---|---|---|---|---|---|
| | | | | DEV | 1 | | DDC4000_#01 | DDC4000 |
| | | | | FIL | 5 | | /01/000/00/00/SY_ | Projektierung |
| | | | | MO | 129 | Bereit | /01/000/00/00/SY_ | Anweisung |
| | | | | MI | 131 | Bereit | /01/000/00/00/SY_ | Zustand |
| | | | | AI | 133 | 100.00 | /01/000/00/00/SY_ | Fortschritt |
| | | | | MI | 135 | kein Fehler | /01/000/00/00/SY_ | Bericht |
| | | | | AV | 137 | 97.00 | /01/000/00/00/SY_ | Restspeicher |
| | | | | FIL | 414 | | /01/000/00/00/SY_ | Def.-Daten |

Figure 4

When you click the ***BACnet test*** tab in the navigation bar, the plug-in's navigation view (1) with the test plan tree appears on the left, and its detailed view is shown on the right (2) (Figure 5). One or more test plans can be created and displayed in the test plan tree. The control elements at the bottom can be used to execute certain test plan management functions.

The new ***BACnet test*** menu holds the menu items:

- Test plan export… (Testplan-Export…) (see also Ch. 5.3)
- Test plan import… (Testplan-Import…) (see also Ch. 5.3)
- Select tests and test cases (Auswahl von Tests und Test Cases) (see Ch. 4.3.3)

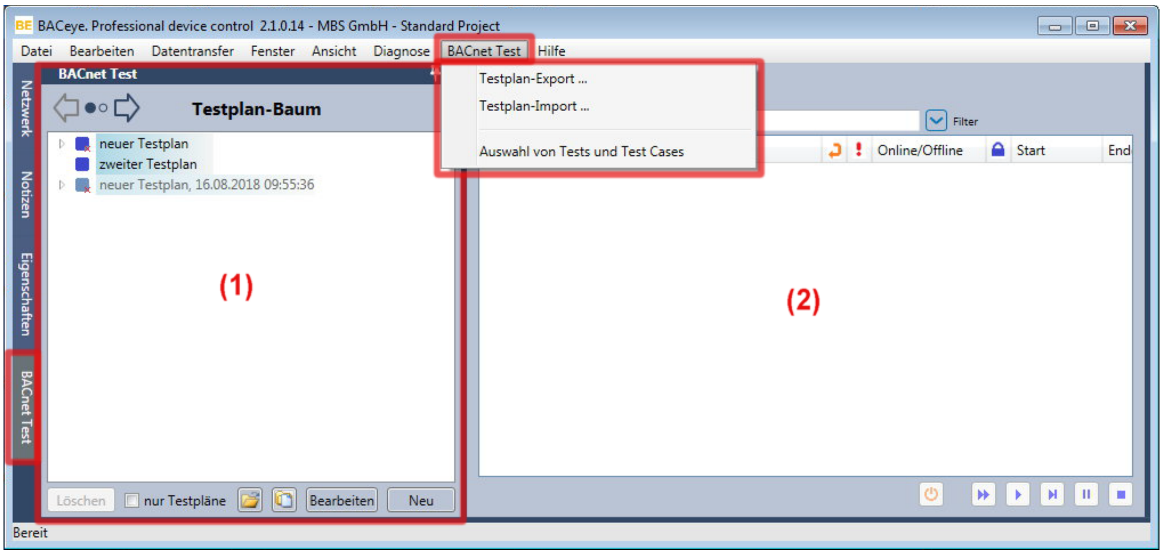
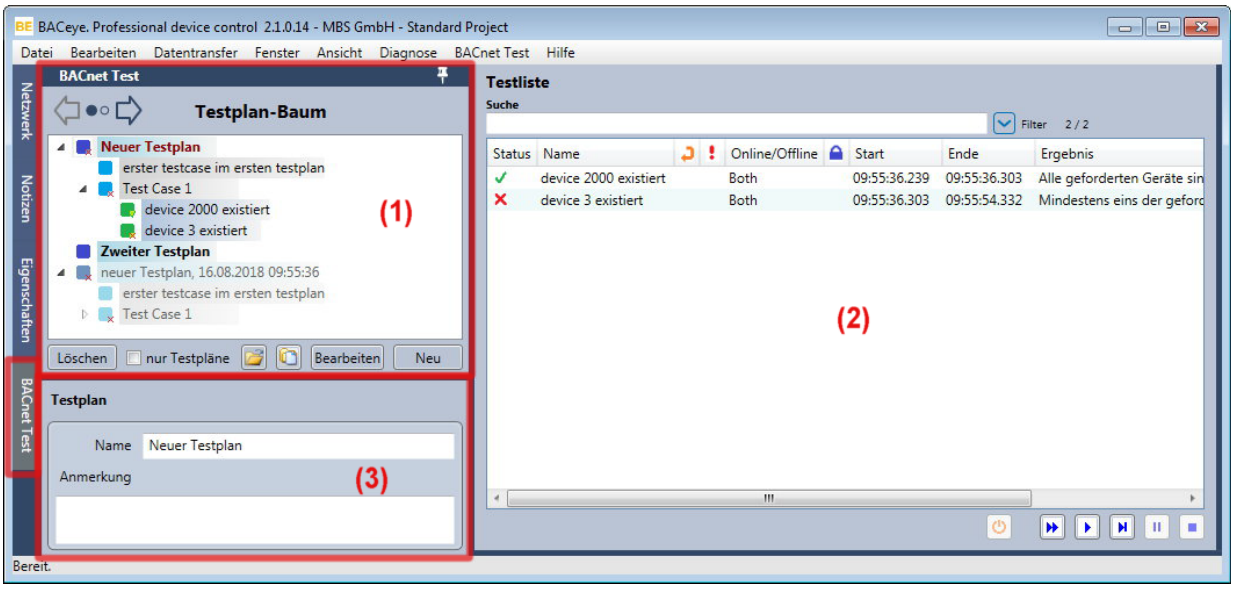


Figure 5

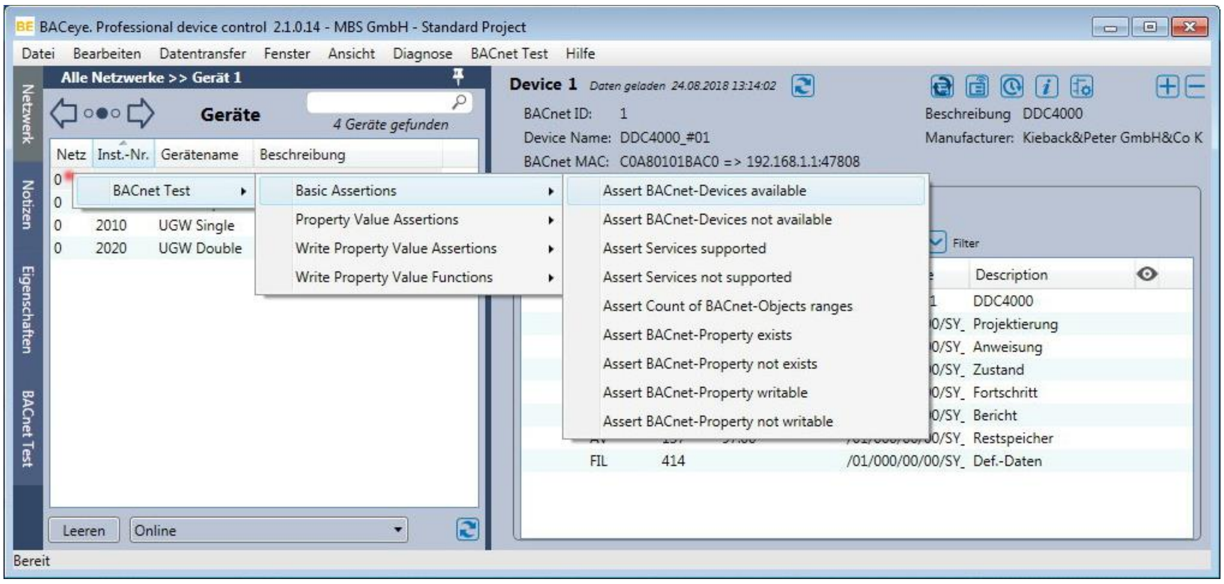
When you select an element (test plan, test case or test/assertion) in the test plan tree, a configuration area for the element (3) selected opens beneath the navigation view (Figure 6).
When a test plan or test case is selected, the list of all the tests that the test plan/ test case includes (test list) is displayed in the detailed view.



| Status | Name | Online/Offline | Start | Ende | Ergebnis |
|---|---|---|---|---|---|
| ✓ | device 2000 existiert | Both | 09:55:36.239 | 09:55:36.303 | Alle geforderten Geräte sin |
| ✗ | device 3 existiert | Both | 09:55:36.303 | 09:55:54.332 | Mindestens eins der geford |

Figure 6

You can also use context menus for structural elements to access the test plug-in's new functions. Figure 7 shows the context menu for a device in the ***Network*** navigation view.


BACeye. Professional device control 2.1.0.14 - MBS GmbH - Standard Project
Datei Bearbeiten Datentransfer Fenster Ansicht Diagnose BACnet Test Hilfe
Alle Netzwerke >> Gerät 1
Geräte
4 Geräte gefunden
Netz Inst.-Nr. Gerätename Beschreibung
BACnet Test
Basic Assertions
Property Value Assertions
Write Property Value Assertions
Write Property Value Functions
Assert BACnet-Devices available
Assert BACnet-Devices not available
Assert Services supported
Assert Services not supported
Assert Count of BACnet-Objects ranges
Assert BACnet-Property exists
Assert BACnet-Property not exists
Assert BACnet-Property writable
Assert BACnet-Property not writable
Leeren
Online
Bereit
Netzwerk
Notizen
Eigenschaften
BACnet Test
Filter

Device 1 Daten geladen 24.08.2018 13:14:02

BACnet ID: 1
Device Name: DDC4000_#01
BACnet MAC: C0A80101BAC0 => 192.168.1.1:47808
Beschreibung DDC4000
Manufacturer: Kieback&Peter GmbH&Co K

| Netz | Inst.-Nr. | Gerätename |
|---|---|---|
| 0 | | |
| 0 | | |
| 0 | 2010 | UGW Single |
| 0 | 2020 | UGW Double |

| | | | Description |
|---|---|---|---|
| | | 1 | DDC4000 |
| | | 0/SY_ | Projektierung |
| | | 0/SY_ | Anweisung |
| | | 0/SY_ | Zustand |
| | | 0/SY_ | Fortschritt |
| | | 0/SY_ | Bericht |
| | | 0/SY_ | Restspeicher |
| FIL | 414 | /01/000/00/00/SY_ | Def.-Daten |

Figure 7

## 3.2 Components

The test plan tree has three branch levels and contains the following hierarchically-divided elements (Figure 8):

1. Test plans
2. Instances of test cases
3. Instances of assertions (tests)

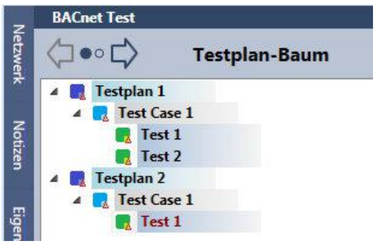


Figure 8

### 3.2.1 Test plan

A test plan is a collection of test cases. Configured test plans can be executed as test runs. Every test plan comprises at least one test case. Every test case has a title and description and can be saved in a file and reloaded from that file.

Context menu for a test plan element in the test plan tree:

| Function | Description |
|---|---|
| New | Creates a new test case |
| Import test case | Imports a test case from a selectable file |
| Empty | Deletes all the test cases in the test plan |
| Delete | Deletes the test plan |
| Export | Exports the test plan to a selectable file |
| Reduce all | Hides the structure that can be seen under this test plan |

**Toolbar for the test plan tree**

Below the test plan tree with the test plans, there is a row with control elements for the following functions:

| Function | Type | Description |
|---|---|---|
| Delete | Button | Deletes the element selected |
| Test plans only | Checkbox | Hide test runs in the test plan tree |
| Export | Button/symbol | Exports test plans or test runs in a selectable file |
| Import | Button/symbol | Imports test plans or test runs from a selectable file |

|  |  |  |
|---|---|---|
| Edit | Button | Opens the ***Select tests and test cases*** (Auswahl von Tests und Test Cases) dialog |
| New | Button | Creates a new test plan |

**Test plan editor**

The test plan editor integrated into the plug-in enables test plans to be created. Users may select pre-defined assertions and test cases. However, they may also create and save their own test cases. Test plans and test cases can also be edited in their saved form (XML).

**Test run**

A test run is the full or partial execution of a test plan.  For every assertion that is executed in the test run, the result, the time and the log entry returned by the assertion are documented. A test run can only be executed once. Each additional execution results in a new test run. A test run's results can be saved, loaded and displayed.

### 3.2.2 Test case

A test case is a grouping of assertions (tests). Every test case has a title and description. However, you may also define your own test cases. Test cases can be saved as a file.

Context menu for a test case element in the test plan tree:

| Function | Description |
|---|---|
| Empty | Deletes all the tests in the test case |
| Delete | Deletes the test case |
| Export | Exports the test case to a selectable file |
| Reduce all | Hides the structure that can be seen under this test case |

### 3.2.3 Assertion (test)

An assertion is a pre-defined test routine (template) that comes with BACeye in an assertion library. The assertion becomes a test when it is added to a test case (see Ch. 4.3.3). An assertion has a set of possibly optional parameters that control the execution. As a self-contained test step, an assertion always has a Boolean return value (True, False) and a textual notification of the test details. When a test is selected in the test plan tree, the test's properties are displayed in the configuration area. The name of the test, or its parameters, can be changed here.

Context menu for a test element in the test plan tree:

| Function | Description |
|---|---|
| Delete | Deletes the test |

**Assertions library**

An assertions library provides access to a set of assertions.  The totality of the assertion libraries installed with the plug-in defines the scope of the available assertions. An assertions library is a .NET library that exports a defined interface via MEF (Managed Extensibility Framework).

**Drag-and-drop in the test plan tree**

You can move or copy the components within the test plan tree by dragging and dropping.  To copy, you simultaneously use the Ctrl key.

Moving tests or test cases in the test plan defines the sequence in which they are run. When test plans are moved or created in the test plan tree, it is only their order in the tree which is changed or defined. A test plan is created behind all the other test plans if its target is a free area in the test plan tree, i.e. not a test plan or one of its components.

# 4 An example as an introduction

## 4.1 Workflow

The plug-in is supplied with a set of test cases and assertions. You can use the test plan editor or edit the XML file to create your own test cases and test plans based on the supplied test cases and assertions.

The following workflow is available for the BACnet test:

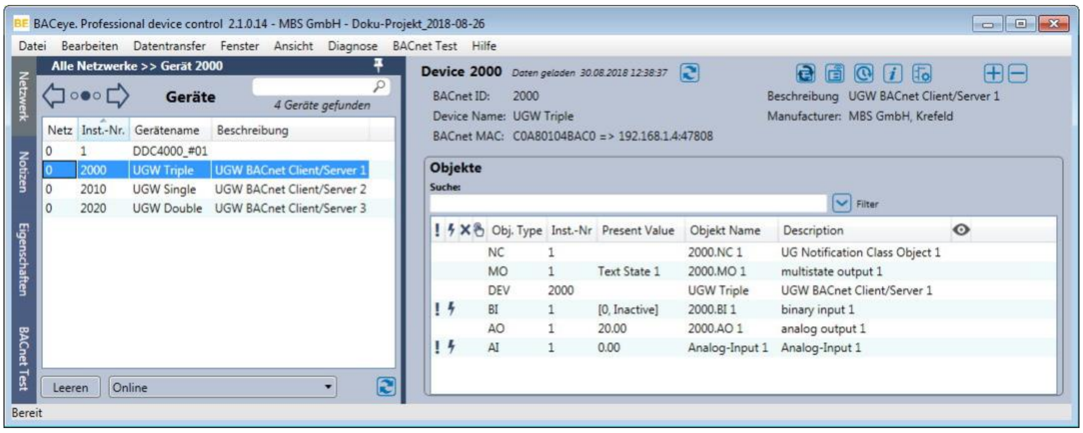
1. Create a new test plan as a container for test cases
2. Define one or more test cases in the test plan as a container/s for tests
3. Define at least one test/assertion in the test case
4. Select a test plan to run
5. Start a test run
6. Analyse the test results

This BACnet test workflow is possible in both BACeye's online and offline mode.

## 4.2 BACnet devices in the network (example of online test)

In the following example, the BACnet network includes the following BACnet devices (Figure 9):

- An automation station DDC4000 (BACnet Device-ID: 1)
- A MBS Universal Gateway UGW Triple (BACnet Device-ID: 2000)
- A MBS Universal Gateway UGW Single (BACnet Device-ID: 2010)
- A MBS Universal Gateway UGW Double (BACnet Device-ID: 2020).



| Netz | Inst.-Nr. | Gerätename | Beschreibung |
|---|---|---|---|
| 0 | 1 | DDC4000_#01 | |
| 0 | 2000 | UGW Triple | UGW BACnet Client/Server 1 |
| 0 | 2010 | UGW Single | UGW BACnet Client/Server 2 |
| 0 | 2020 | UGW Double | UGW BACnet Client/Server 3 |

Device 2000 Daten geladen 30.08.2018 12:38:37

BACnet ID: 2000
Device Name: UGW Triple
BACnet MAC: C0A80104BAC0 => 192.168.1.4:47808
Beschreibung UGW BACnet Client/Server 1
Manufacturer: MBS GmbH, Krefeld

| ! | ϟ | Obj. Type | Inst.-Nr | Present Value | Objekt Name | Description |
|---|---|---|---|---|---|---|
| | | NC | 1 | | 2000.NC 1 | UG Notification Class Object 1 |
| | | MO | 1 | Text State 1 | 2000.MO 1 | multistate output 1 |
| | | DEV | 2000 | | UGW Triple | UGW BACnet Client/Server 1 |
| ! | ϟ | BI | 1 | [0, Inactive] | 2000.BI 1 | binary input 1 |
| | | AO | 1 | 20.00 | 2000.AO 1 | analog output 1 |
| ! | ϟ | AI | 1 | 0.00 | Analog-Input 1 | Analog-Input 1 |

Figure 9

## 4.3 Use the test plan editor

### 4.3.1 Create a new test plan

Activate the ***BACnet test*** tab in the navigation bar.

- Create a new, empty test plan by clicking **[New]** (Neu) in the test plan tree toolbar. This generates a numbered test plan. Beneath the test plan tree, the configuration area opens with the test plan properties (Figure 10).
- Accept the name the program proposes or enter a new, unique name for the test plan in the configuration area. You have the option of adding some explanatory text about this test plan in the *Comment* (Anmerkung) text field.

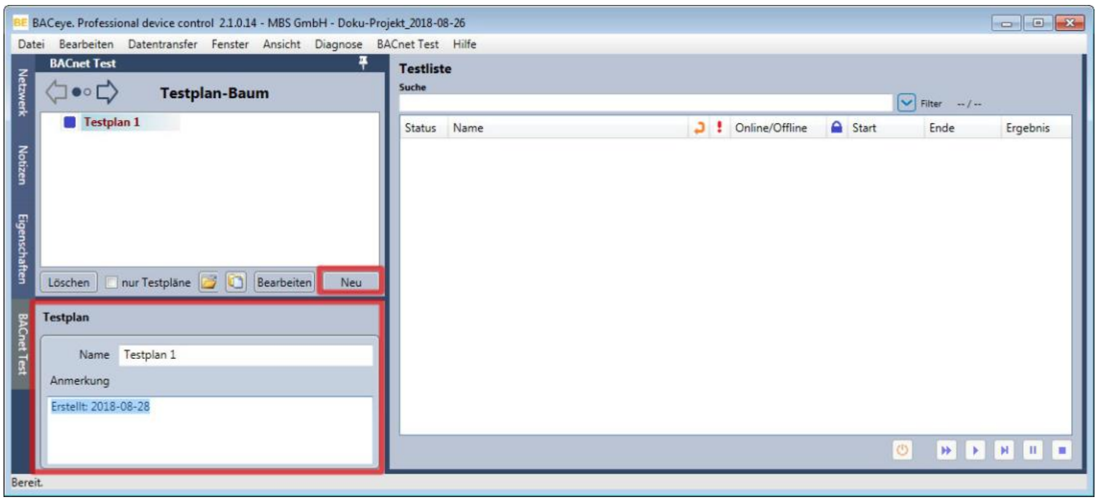

BACeye. Professional device control 2.1.0.14 - MBS GmbH - Doku-Projekt_2018-08-26
Datei Bearbeiten Datentransfer Fenster Ansicht Diagnose BACnet Test Hilfe
BACnet Test
Testplan-Baum
Testplan 1
Löschen
nur Testpläne
Bearbeiten
Neu
Testplan
Name Testplan 1
Anmerkung
Erstellt: 2018-08-28
Testliste
Suche
Filter
Status Name Online/Offline Start Ende Ergebnis
Bereit.
Netzwerk
Notizen
Eigenschaften
BACnet Test

Figure 10

**Note:**
You can delete the test plan again by clicking the **[Delete]** (Löschen) button or selecting *Delete* (Löschen) from the context menu. Select **[OK]** to confirm. This applies to test cases, too.

## 4.3.2 Create a new test case

- You create a new test case as the second structural level by clicking ***New*** (Neu) in the test plan context menu (Figure 11Figure 10). You can also use the ***Import test case*** (Testcase importieren) menu item in the test plan context menu to import a test case from a selectable file.

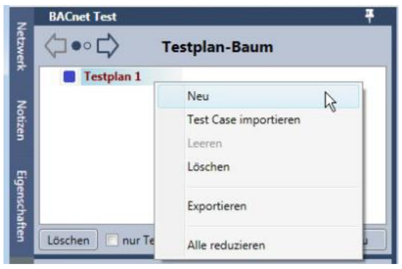


Figure 11

- Accept the name the program proposes or enter a new, unique name for the test case in the configuration area (Figure 12). You have the option of adding some explanatory text about this test case in the *Comment* (Anmerkung) text field.

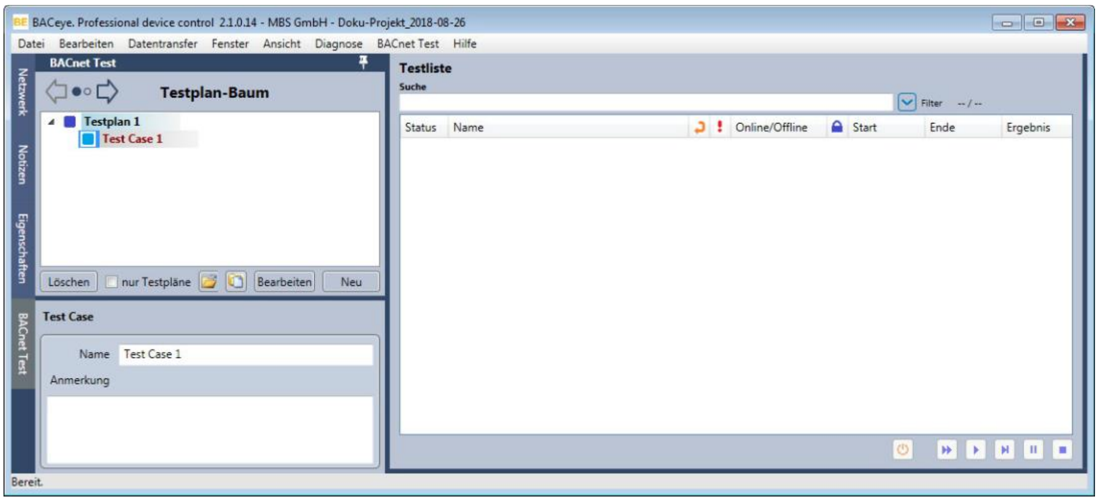


Figure 12

## 4.3.3 Create and configure a new test

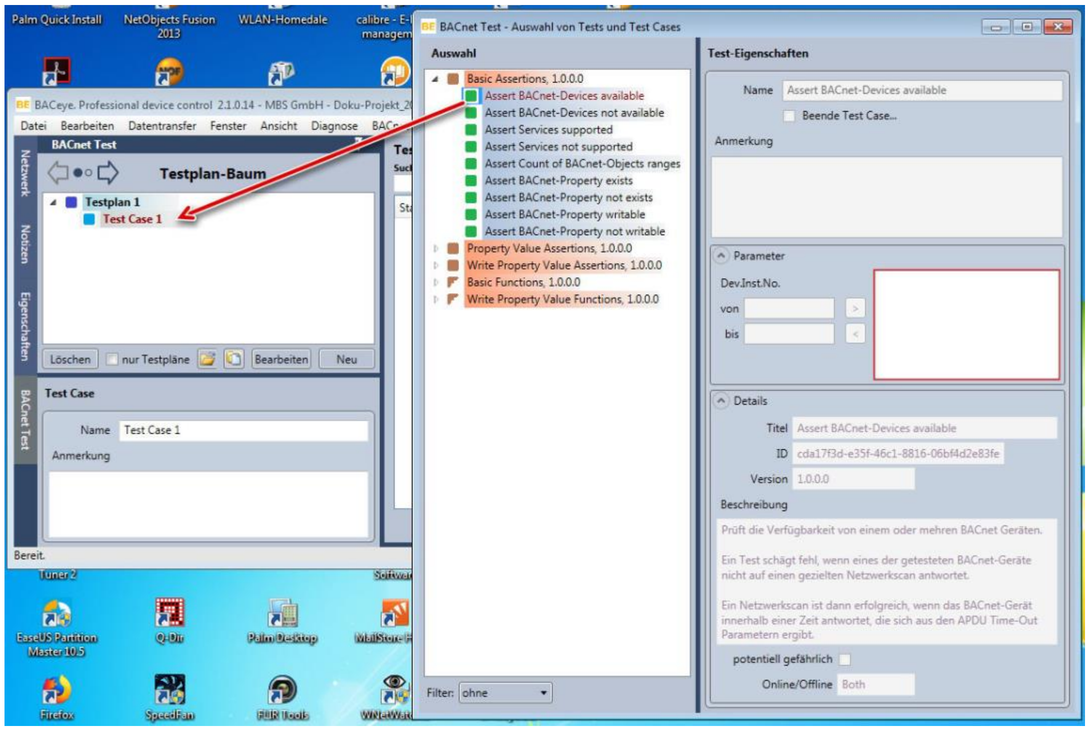
- You create a new test to be the third structural level by clicking **[Edit]** (Bearbeiten) in the test plan tree toolbar. Alternatively, you can use the menu path***BACnet test > Select tests and test cases.*** (BACnet Test > Auswahl Tests und Test Cases) The ***Select tests and test cases*** (Auswahl Tests und Test Cases) window then opens (Figure 13).

The ***Select tests and test cases*** (Auswahl Tests und Test Cases) window contains a pre-defined range of test templates (assertions) . The area on the left shows a drop-down structure tree of different assertion libraries and the pre-defined assertions and test cases they contain. The area on the right shows the properties of the element selected in the structure tree.

- Drop down the assertion library in the structure tree of the ***Select tests and test cases*** (Auswahl Tests und Test Cases) window, and select the assertion you want.

- Drag the selected assertion to the test plan tree and drop it onto the test case that will hold the test (Figure 13).

When this is done, instances or copies are generated from the relevant templates and entered into the test plan. The test is added below the test cases and displayed in the test list (Figure 14).


BACnet Test - Auswahl von Tests und Test Cases
Auswahl
Basic Assertions, 1.0.0.0
Assert BACnet-Devices available
Assert BACnet-Devices not available
Assert Services supported
Assert Services not supported
Assert Count of BACnet-Objects ranges
Assert BACnet-Property exists
Assert BACnet-Property not exists
Assert BACnet-Property writable
Assert BACnet-Property not writable
Property Value Assertions, 1.0.0.0
Write Property Value Assertions, 1.0.0.0
Basic Functions, 1.0.0.0
Write Property Value Functions, 1.0.0.0
Filter: ohne
Test-Eigenschaften
Name
Beende Test Case...
Anmerkung
Parameter
Dev.Inst.No.
von
bis
Details
Titel
ID
Version
Beschreibung
potentiell gefährlich
Online/Offline
BACeye. Professional device control 2.1.0.14 - MBS GmbH - Doku-Projekt_2
Datei Bearbeiten Datentransfer Fenster Ansicht Diagnose
BACnet Test
Testplan-Baum
Testplan 1
Test Case 1
Löschen
nur Testpläne
Bearbeiten
Neu
Test Case
Name
Anmerkung
Bereit.

Name: Assert BACnet-Devices available

Titel: Assert BACnet-Devices available

ID: cda17f3d-e35f-46c1-8816-06bf4d2e83fe

Version: 1.0.0.0

Beschreibung: Prüft die Verfügbarkeit von einem oder mehren BACnet Geräten.

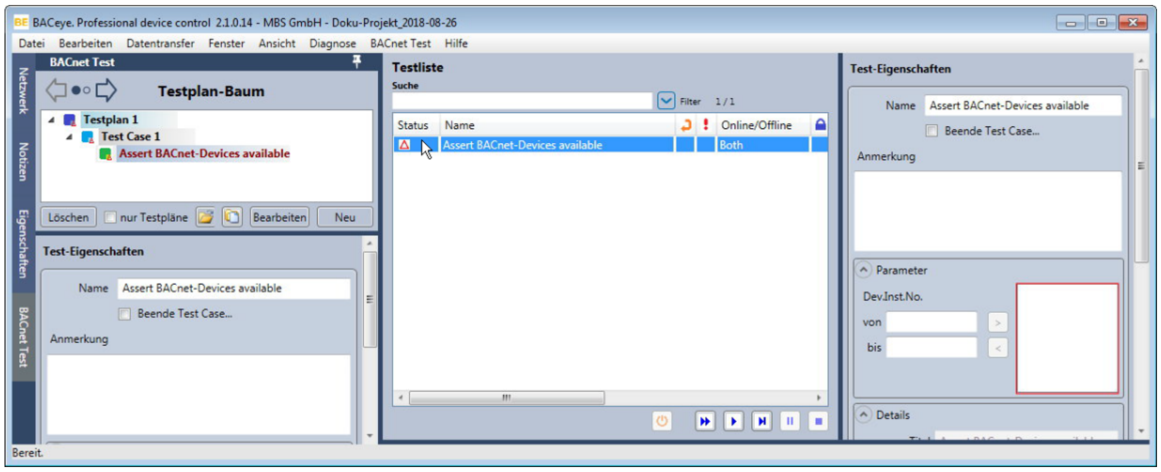
Ein Test schägt fehl, wenn eines der getesteten BACnet-Geräte nicht auf einen gezielten Netzwerkscan antwortet.

Ein Netzwerkscan ist dann erfolgreich, wenn das BACnet-Gerät innerhalb einer Zeit antwortet, die sich aus den APDU Time-Out Parametern ergibt.

Online/Offline: Both

Name: Test Case 1

Figure 13



| Status | Name | Online/Offline |
|---|---|---|
| △ | Assert BACnet-Devices available | Both |

Figure 14

In this specific case, (Figure 13 and Figure 14), the assertion *Assert BACnet-Devices available* was dragged from the *Basic Assertion 1.0.0.0* assertion library and dropped into Test Case 1. This assertion enables the availability of one or more BACnet devices in the BACnet network to be tested. The instances of this assertion ensure, for subsequent tests, that all the necessary BACnet devices are present in the BACnet network and entered in BACeye.

## Test list

The test list shows the list of tests that is assigned to the element currently selected in the test plan tree. The test status output to the first column in the test list differentiates between the following statuses:

| Status | Symbol | Description |
|---|---|---|
| Missing parameter | △ | The test has not been parametrised or has been insufficiently parametrised. |
| Deactivated | ○ | The test is deactivated. |
| No result | ● | Test was not run. |
| Test is being run | ↻ | Test is being run. |

| | | |
|---|---|---|
| Error | ✗ | At least one error occurred during the test run. |
| Successful | ✓ | No errors occurred during the test run. |

**Test properties**

If you select the test that was added to the test list, the test's properties are displayed in the configuration area to the right. If the test has not been pre-defined, input fields that are missing information are outlined in red.

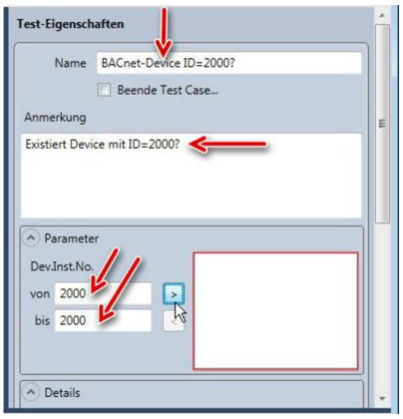
The test properties area has the following input fields and editing areas:

| Property | Type | Description |
|---|---|---|
| Name | Input field | Name of the test<br>Pre-populated with the title of the assertion. |
| Comment | Text field | Comment about the test<br>Pre-populated with the description of the assertion. |
| Parameter | Area that can be shown and hidden | This is where the user control is shown that provides the assertion so that its test-specific parameters can be edited. |
| Details | Area that can be shown and hidden | This shows all the properties of the assertion that the test instance has been created from: title, GUID, version, description, risk potential, online/offline |

- To modify the name of the test, enter the new name in the *Name* input field.
- You have the option of adding a descriptive comment in the *Comment* (Anmerkung) text field.
- Enter the device ID you want in the*Dev.Inst.No.* fields in the *Parameters* (Parameter) area (here: 2000).
- Copy the ID value with the button to the field outlined in red (Figure 15).

This simple test has now been parametrised and is ready for the test run (Figure 16).

To illustrate a case in which a BACnet device is not found in the network, another test was generated from the *Assert BACnet-Devices available* assertion and added to the sample test project. Its aim is to test whether a BACnet device with the ID = 2030 is present in the network (Figure 17).



Test-Eigenschaften

Name: BACnet-Device ID=2000?

☐ Beende Test Case...

Anmerkung: Existiert Device mit ID=2000?

Parameter

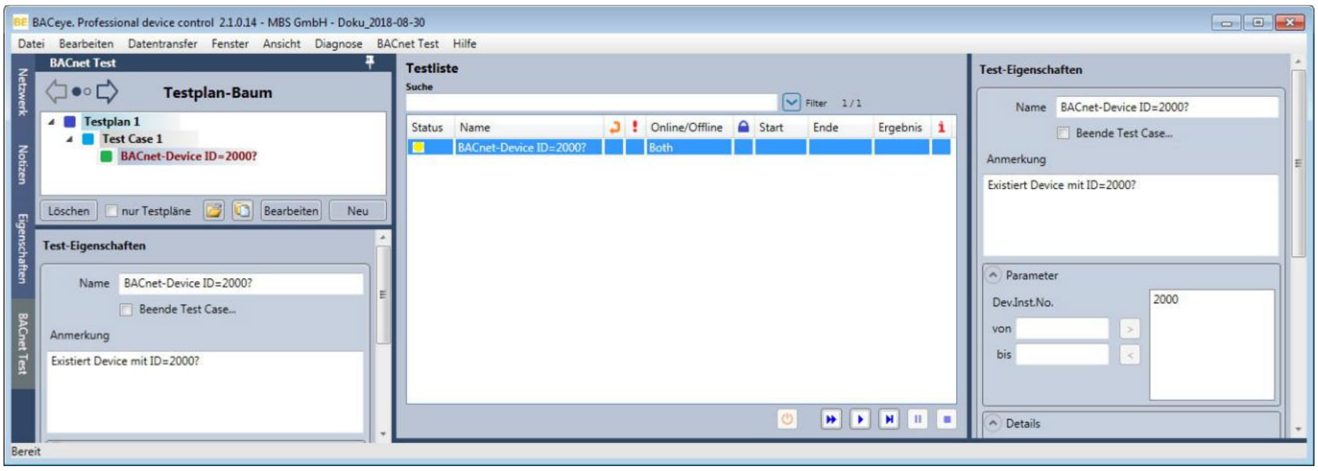
Dev.Inst.No.

von: 2000

bis: 2000

Details

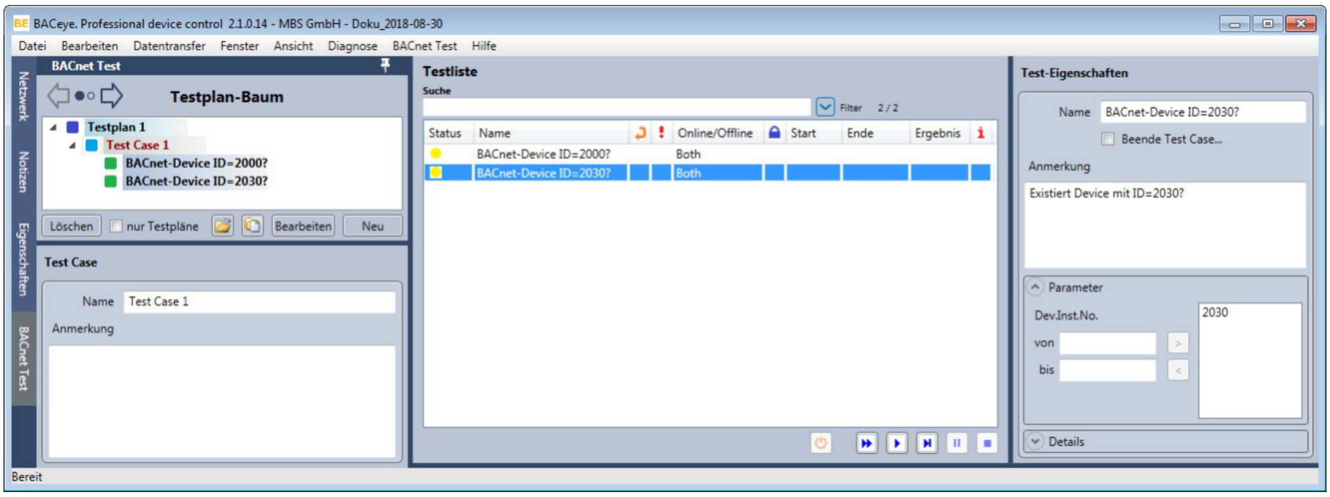
Figure 15


BACeye. Professional device control 2.1.0.14 - MBS GmbH - Doku_2018-08-30
Datei Bearbeiten Datentransfer Fenster Ansicht Diagnose BACnet Test Hilfe
BACnet Test
Testplan-Baum
Testplan 1
Test Case 1
BACnet-Device ID=2000?
Löschen nur Testpläne Bearbeiten Neu
Test-Eigenschaften
Testliste
Suche
Filter 1 / 1
Status Name Online/Offline Start Ende Ergebnis
Parameter
Details
Bereit

| Status | Name | Online/Offline | Start | Ende | Ergebnis |
|---|---|---|---|---|---|
| | BACnet-Device ID=2000? | Both | | | |

Name: BACnet-Device ID=2000?
Beende Test Case...
Anmerkung: Existiert Device mit ID=2000?
Dev.Inst.No.: 2000

Figure 16


BACeye. Professional device control 2.1.0.14 - MBS GmbH - Doku_2018-08-30
Datei Bearbeiten Datentransfer Fenster Ansicht Diagnose BACnet Test Hilfe
BACnet Test
Testplan-Baum
Testplan 1
Test Case 1
BACnet-Device ID=2000?
BACnet-Device ID=2030?
Löschen nur Testpläne Bearbeiten Neu
Test Case
Testliste
Suche
Filter 2 / 2
Test-Eigenschaften
Parameter
Details
Bereit

| Status | Name | Online/Offline | Start | Ende | Ergebnis |
|---|---|---|---|---|---|
| ● | BACnet-Device ID=2000? | Both | | | |
| | BACnet-Device ID=2030? | Both | | | |

Test Case — Name: Test Case 1
Name: BACnet-Device ID=2030?
Beende Test Case...
Anmerkung: Existiert Device mit ID=2030?
Dev.Inst.No.: 2030

Figure 17

### Save changes in the test plan tree

You save changes in the test plan tree by saving the current BACeye project (see Ch. 5). Any components that have been changed or created since the last save or load procedure are flagged in the tree by means of a bold font.

## 4.4 Execute a test run

The test list shows all the tests that have been assigned to the selected component in the test plan tree. The sequence of the tests in the test list is the same as that in the test plan tree and determines the order in which the tests are executed in the test run.

### Control bar for managing the test run

Below the test list there is a control bar with the following buttons:

| Function | Type | Description |
|---|---|---|
| Start | Button ▶ | A test run is started/resumed. |
| Single step | Button ▶\| | Only the first/next test in the test run is executed and there is a pause before the next test is executed. If there are no more tests to execute, the test run ends. |
| Next | Button ⏩ | Skips the next executable test. If it is the last one in the test run, the test run ends. If the test run is currently paused, the next test is skipped and there is a pause before the following test is executed. |

| | | |
|---|---|---|
| Pause | Button ⏸ | Pauses the test run after executing the current test and before the following executable test. If there are no more tests to execute, the test run ends. |
| Stop | Button ■ | After requesting confirmation and after executing the current test, ends the test run. If the test run is currently paused, it ends immediately. |
| Emergency stop | Button ⏻ | Aborts the current test with no analysis and stops the test run. |

### 4.4.1 Start the test run

- To execute all the tests in a test plan, select the relevant test plan in the test plan tree.  If only one particular test is to be executed, select it in the test plan tree so that only that test appears in the test list.

To achieve a particular scope of testing, the tests in the list can also be delimited using the filter and search functions at the top.

- Select the Start button ▶ to start a test run based on the tests in the test list.

**Note:**

You can also execute the tests in the test run one by one using the Single step button ⏭.

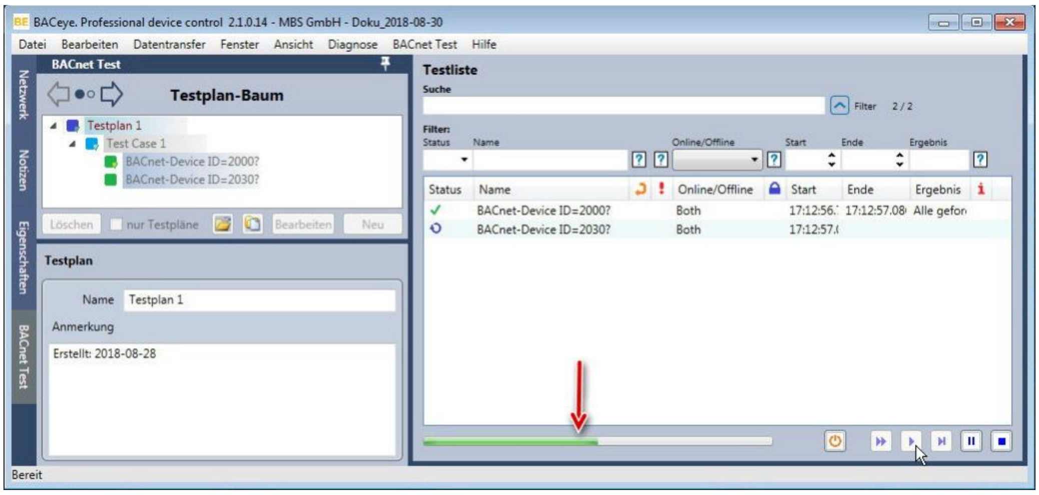
A test run is not started if there are any tests in the list which have not been adequately parametrised (status: △). A corresponding dialog will draw attention to this. You can also use the context menu to deactivate tests in the test list. Deactivated tests are ignored in the test run.

First of all, when a test run is started, any results for all the tests in the test plan from any previous test run are removed.

The tests in the test list are then worked through, one after the other. Progress is indicated by a progress bar below the test list (Figure 18).

**Note:**

You can use the Pause button ⏸ to pause the test run. If this is done, the next executable test is flagged with the status symbol ⏸. You can resume a paused test with the Start button ▶ or resume it in stages with the Single step button ⏭. The Stop button ■ will end a test run prematurely. If this is done, the test results acquired are retained.



BACeye. Professional device control 2.1.0.14 - MBS GmbH - Doku_2018-08-30

| Status | Name | Online/Offline | Start | Ende | Ergebnis |
|---|---|---|---|---|---|
| ✓ | BACnet-Device ID=2000? | Both | 17:12:56. | 17:12:57.08 | Alle gefor |
| | BACnet-Device ID=2030? | Both | 17:12:57.( | | |

Testplan

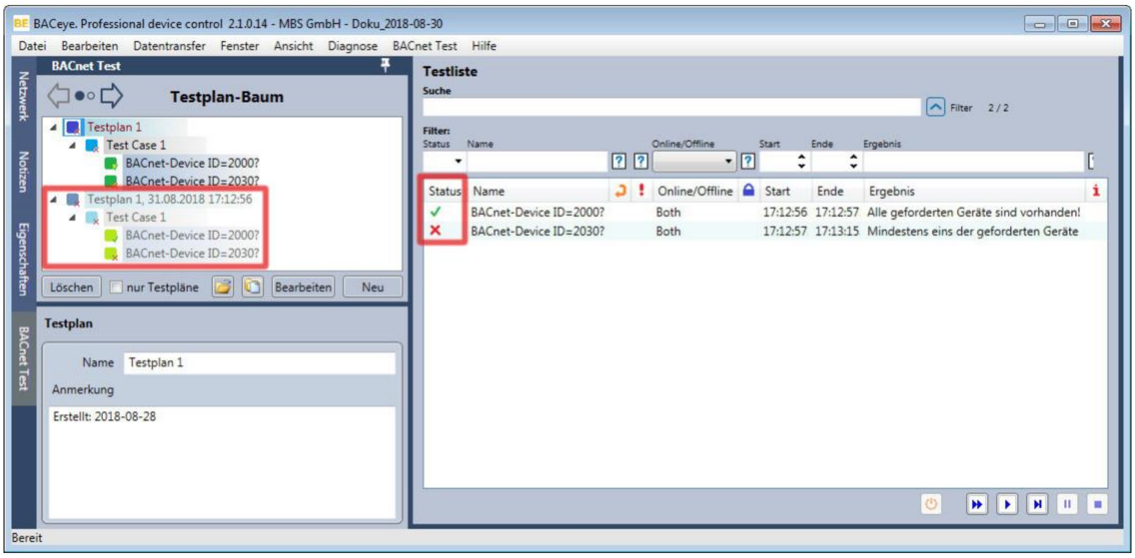
Name: Testplan 1

Anmerkung: Erstellt: 2018-08-28

Figure 18

## 4.4.2 Display test results

While a test is running, the status of the individual test updates in the test list to reflect success or failure ( ✗ for an error or ✓ if successful). Start and end times and result texts are displayed (Figure 19).



BACeye. Professional device control 2.1.0.14 - MBS GmbH - Doku_2018-08-30

| Status | Name | Online/Offline | Start | Ende | Ergebnis |
|---|---|---|---|---|---|
| ✓ | BACnet-Device ID=2000? | Both | 17:12:56 | 17:12:57 | Alle geforderten Geräte sind vorhanden! |
| ✗ | BACnet-Device ID=2030? | Both | 17:12:57 | 17:13:15 | Mindestens eins der geforderten Geräte |

Testplan

Name: Testplan 1

Anmerkung: Erstellt: 2018-08-28

Figure 19

- As the BACnet device with ID = 2000 was found in the BACnet network, that test was successful ✓.
- As the BACnet device with ID = 2030 was not found in the BACnet network before the timeout, that test run was unsuccessful ✗.

Response to this test:

✚ If the BACnet device with ID = 2030 is required, add it to the BACnet network.

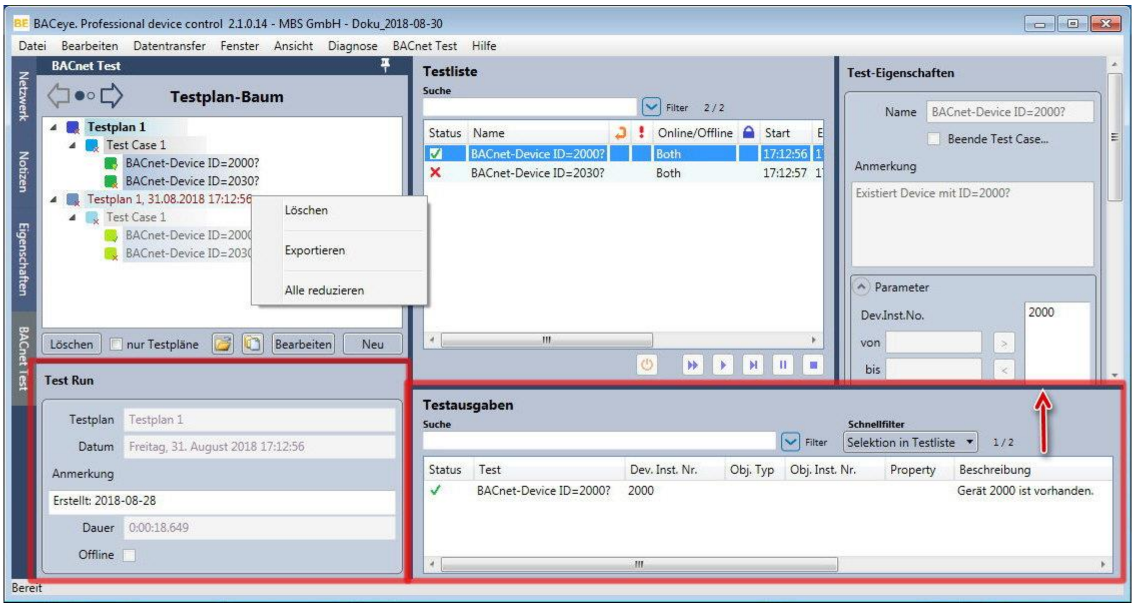
**Other ways of analysing the test run**

In the test plan tree, the test results also display with the symbols ✗ and ✓ by the components.

When a test run ends, a copy of the test plan is added beneath the test plan (in grey text). The components in this copy (= test run) map their originals in the test plan. A test run in the test plan tree is a write-protected copy of the test plan that was executed.

The context menu for a test run in the test plan tree includes these functions (Figure 20):

| Function | Description |
|---|---|
| Delete | Deletes the selected test run (also using the **[Delete]** (Löschen) button) |
| Export | Exports the selected test run in a selectable file |
| Reduce all | Hides the structure that can be seen under this test run |



Testplan 1
Test Case 1
BACnet-Device ID=2000?
BACnet-Device ID=2030?
Testplan 1, 31.08.2018 17:12:56
Test Case 1
BACnet-Device ID=2000
BACnet-Device ID=2030

Testliste: 2 / 2

| Status | Name | Online/Offline | Start |
|---|---|---|---|
| ✓ | BACnet-Device ID=2000? | Both | 17:12:56 |
| ✗ | BACnet-Device ID=2030? | Both | 17:12:57 |

Test-Eigenschaften
Name: BACnet-Device ID=2000?
Beende Test Case...
Anmerkung: Existiert Device mit ID=2000?
Parameter
Dev.Inst.No.: 2000
von
bis

Test Run
Testplan: Testplan 1
Datum: Freitag, 31. August 2018 17:12:56
Anmerkung: Erstellt: 2018-08-28
Dauer: 0:00:18.649
Offline

Testausgaben — Schnellfilter: Selektion in Testliste, 1 / 2

| Status | Test | Dev. Inst. Nr. | Obj. Typ | Obj. Inst. Nr. | Property | Beschreibung |
|---|---|---|---|---|---|---|
| ✓ | BACnet-Device ID=2000? | 2000 | | | | Gerät 2000 ist vorhanden. |

Figure 20

The configuration area for the test run beneath the test plan tree contains the following properties:

| Property | Type | Description |
|---|---|---|
| Test plan | Input field | Name of the test plan on which the test run was executed (non-editable) |
| Date | Input field | Time and date the test run was executed (non-editable) |
| Comment | Input field | Optional comment about the test plan on which the test run was executed (editable). |
| Duration | Input field | Either the duration or the end time of the test run (non-editable) |
| Offline | Checkbox | Indicates whether the test was run offline (non-editable) |
| Snapshot | Input field | If the test was run offline, the snapshot used is specified here (non-editable). |

When a test run component is selected in the test plan tree, the list of associated tests appears in the detailed view, just as it does for the test plan.

In the lower part of the detailed view, separated by a dark blue line, a list of the tested components is shown in the *Test outputs* (Testausgaben) area (Figure 20). The test result is shown for each component.

**Note:**
If the *Test outputs* (Testausgaben) area is not visible, drag the dark blue separator line upwards and drop it (see Figure 20).

## 4.5 Execute a test run offline

If snapshots have been imported into the current BACeye project (see BACeye manual), a test on one of these snapshots can be run offline. To do this, the snapshot to be used must be selected in BACeye.

If a test is run offline, only tests that can be run offline are executed. All the others are skipped, e.g. a test that attempts to write about BACnet.

When the test run ends, the fact that it was run offline and which snapshot was used are also stored.

## 4.6 Examples of other, more complex test plans

After testing for the required BACnet devices, you can then use the BACnet-Test plug-in to test other BACnet properties.
For example, test for the following properties with separate test plans (Figure 21):

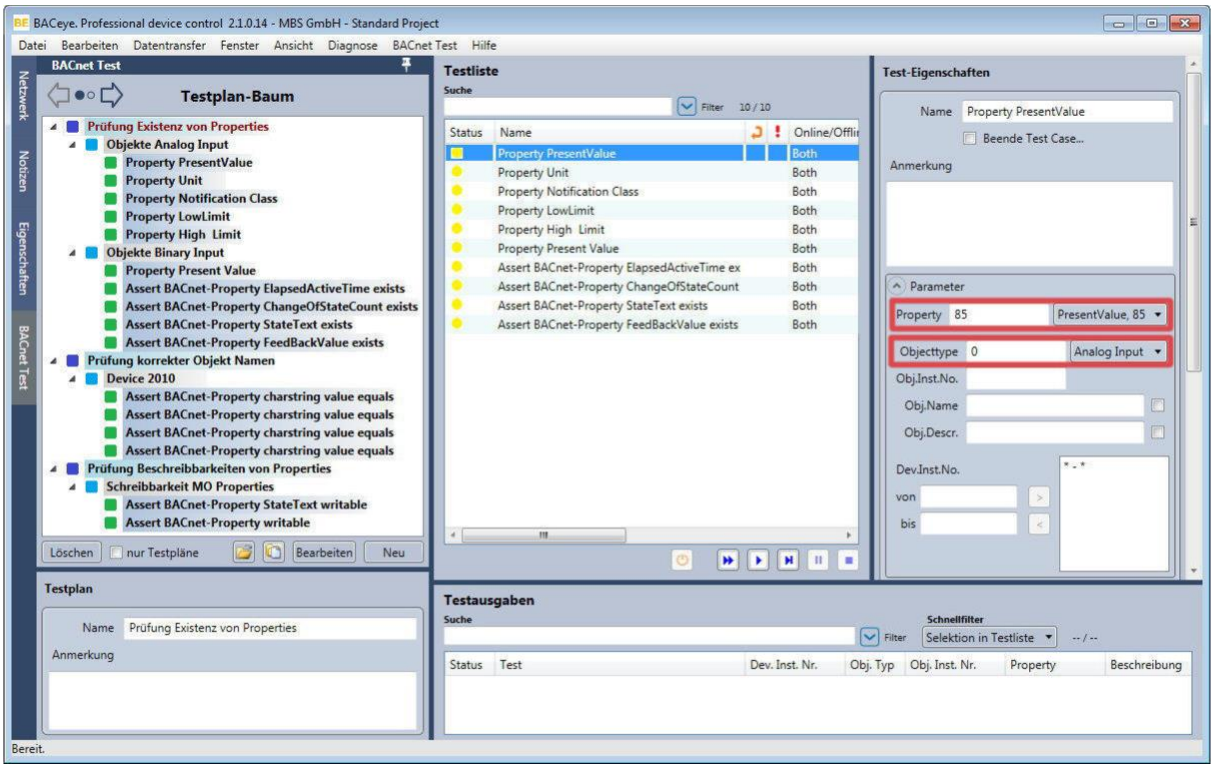
- Existence of properties
- Correct object name
- Describabilities of properties

To do this, create the individual test plans and test cases, as shown above. To test for existing properties, you can divide into two test cases with tests for objects with analogue and digital input (see Figure 21).

You get the assertion templates for the tests from the following assertion libraries:

- Basic Assertions (Tests *Assert BACnet-Property exists*, *Assert BACnet-Property writable*)
- Property Value Assertions (Test *Assert BACnet-Property charstring value equals*)

Configure the test properties you require in the detailed view on the right, e.g. the precise property to be tested and the object type in the *Parameters* (Parameter) area (see Figure 21).



BACeye. Professional device control 2.1.0.14 - MBS GmbH - Standard Project

BACnet Test

Testplan-Baum

- Prüfung Existenz von Properties
  - Objekte Analog Input
    - Property PresentValue
    - Property Unit
    - Property Notification Class
    - Property LowLimit
    - Property High Limit
  - Objekte Binary Input
    - Property Present Value
    - Assert BACnet-Property ElapsedActiveTime exists
    - Assert BACnet-Property ChangeOfStateCount exists
    - Assert BACnet-Property StateText exists
    - Assert BACnet-Property FeedBackValue exists
- Prüfung korrekter Objekt Namen
  - Device 2010
    - Assert BACnet-Property charstring value equals
    - Assert BACnet-Property charstring value equals
    - Assert BACnet-Property charstring value equals
    - Assert BACnet-Property charstring value equals
- Prüfung Beschreibbarkeiten von Properties
  - Schreibbarkeit MO Properties
    - Assert BACnet-Property StateText writable
    - Assert BACnet-Property writable

Testplan

Name: Prüfung Existenz von Properties

Anmerkung

Testliste

Suche

10 / 10

| Status | Name | Online/Offli |
|---|---|---|
| ● | Property PresentValue | Both |
| ● | Property Unit | Both |
| ● | Property Notification Class | Both |
| ● | Property LowLimit | Both |
| ● | Property High Limit | Both |
| ● | Property Present Value | Both |
| ● | Assert BACnet-Property ElapsedActiveTime ex | Both |
| ● | Assert BACnet-Property ChangeOfStateCount | Both |
| ● | Assert BACnet-Property StateText exists | Both |
| ● | Assert BACnet-Property FeedBackValue exists | Both |

Test-Eigenschaften

Name: Property PresentValue

Beende Test Case...

Anmerkung

Parameter

Property: 85 — PresentValue, 85

Objecttype: 0 — Analog Input

Obj.Inst.No.

Obj.Name

Obj.Descr.

Dev.Inst.No. — * - *

von

bis

Testausgaben

Suche

-- / --

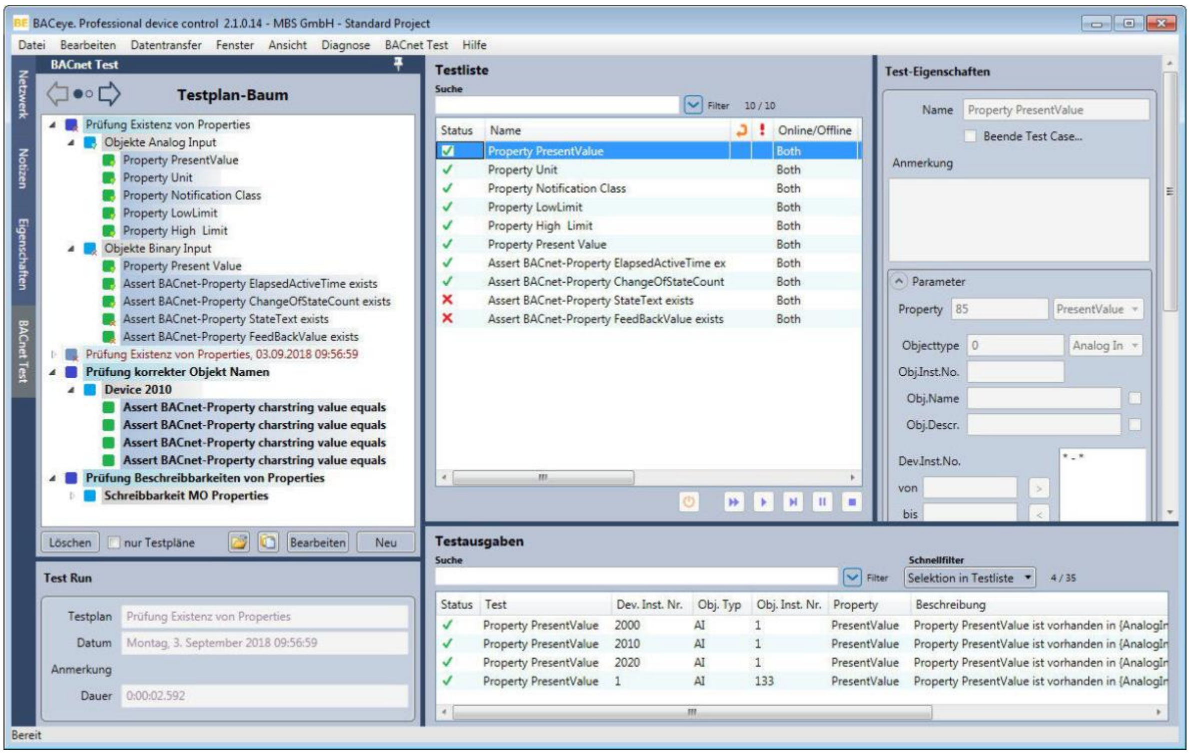
| Status | Test | Dev. Inst. Nr. | Obj. Typ | Obj. Inst. Nr. | Property | Beschreibung |
|---|---|---|---|---|---|---|

Figure 21

**Note:**
For further information on the assertions supplied to date and on their properties, please refer to the MBS document „Spezifikation - BACnet-Test Plug-In Assertions.pdf“.

Use the Start button to start the three tests one after the other.

The result, e.g. for the first test run, shows Figure 22. The *Test outputs* (Testausgaben) area lists the test results for each test component.



BACeye. Professional device control 2.1.0.14 - MBS GmbH - Standard Project

BACnet Test

Testplan-Baum

- Prüfung Existenz von Properties
  - Objekte Analog Input
    - Property PresentValue
    - Property Unit
    - Property Notification Class
    - Property LowLimit
    - Property High Limit
  - Objekte Binary Input
    - Property Present Value
    - Assert BACnet-Property ElapsedActiveTime exists
    - Assert BACnet-Property ChangeOfStateCount exists
    - Assert BACnet-Property StateText exists
    - Assert BACnet-Property FeedBackValue exists
- Prüfung Existenz von Properties, 03.09.2018 09:56:59
- **Prüfung korrekter Objekt Namen**
  - **Device 2010**
    - **Assert BACnet-Property charstring value equals**
    - **Assert BACnet-Property charstring value equals**
    - **Assert BACnet-Property charstring value equals**
    - **Assert BACnet-Property charstring value equals**
- **Prüfung Beschreibbarkeiten von Properties**
  - **Schreibbarkeit MO Properties**

Test Run

| | |
|---|---|
| Testplan | Prüfung Existenz von Properties |
| Datum | Montag, 3. September 2018 09:56:59 |
| Anmerkung | |
| Dauer | 0:00:02.592 |

Testliste

Suche — Filter 10 / 10

| Status | Name | Online/Offline |
|---|---|---|
| ✓ | Property PresentValue | Both |
| ✓ | Property Unit | Both |
| ✓ | Property Notification Class | Both |
| ✓ | Property LowLimit | Both |
| ✓ | Property High Limit | Both |
| ✓ | Property Present Value | Both |
| ✓ | Assert BACnet-Property ElapsedActiveTime ex | Both |
| ✓ | Assert BACnet-Property ChangeOfStateCount | Both |
| ✗ | Assert BACnet-Property StateText exists | Both |
| ✗ | Assert BACnet-Property FeedBackValue exists | Both |

Test-Eigenschaften

| | |
|---|---|
| Name | Property PresentValue |
| | Beende Test Case... |
| Anmerkung | |

Parameter

| | | |
|---|---|---|
| Property | 85 | PresentValue |
| Objecttype | 0 | Analog In |
| Obj.Inst.No. | | |
| Obj.Name | | |
| Obj.Descr. | | |
| Dev.Inst.No. | | *-* |
| von | | |
| bis | | |

Testausgaben

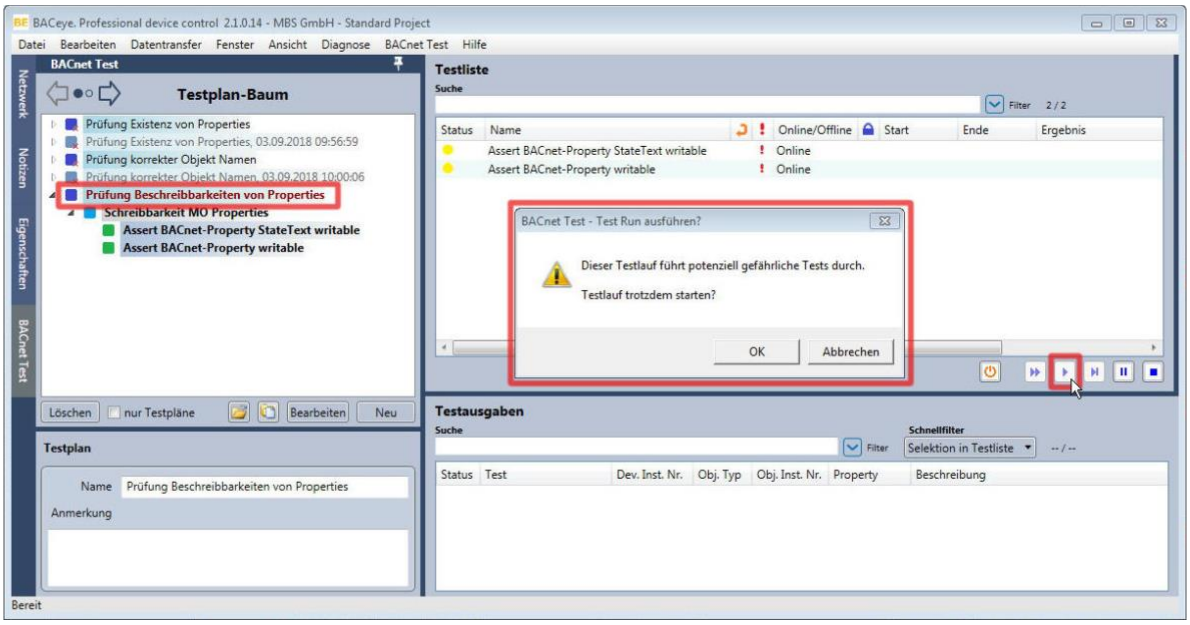
Suche — Filter — Schnellfilter: Selektion in Testliste — 4 / 35

| Status | Test | Dev. Inst. Nr. | Obj. Typ | Obj. Inst. Nr. | Property | Beschreibung |
|---|---|---|---|---|---|---|
| ✓ | Property PresentValue | 2000 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogIn |
| ✓ | Property PresentValue | 2010 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogIn |
| ✓ | Property PresentValue | 2020 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogIn |
| ✓ | Property PresentValue | 1 | AI | 133 | PresentValue | Property PresentValue ist vorhanden in {AnalogIn |

Bereit

Figure 22

When you start tests for the describability of BACnet elements (e.g. of properties), a dialog asks for confirmation that you want to run these tests (Figure 23).



BACeye. Professional device control 2.1.0.14 - MBS GmbH - Standard Project

BACnet Test

Testplan-Baum

- Prüfung Existenz von Properties
- Prüfung Existenz von Properties, 03.09.2018 09:56:59
- Prüfung korrekter Objekt Namen
- Prüfung korrekter Objekt Namen, 03.09.2018 10:00:06
- **Prüfung Beschreibbarkeiten von Properties**
  - **Schreibbarkeit MO Properties**
    - **Assert BACnet-Property StateText writable**
    - **Assert BACnet-Property writable**

Testplan

| | |
|---|---|
| Name | Prüfung Beschreibbarkeiten von Properties |
| Anmerkung | |

Testliste

Suche — Filter 2 / 2

| Status | Name | ! | Online/Offline | Start | Ende | Ergebnis |
|---|---|---|---|---|---|---|
| ● | Assert BACnet-Property StateText writable | ! | Online | | | |
| ● | Assert BACnet-Property writable | ! | Online | | | |

BACnet Test - Test Run ausführen?

Dieser Testlauf führt potenziell gefährliche Tests durch.

Testlauf trotzdem starten?

Testausgaben

Suche — Filter — Schnellfilter: Selektion in Testliste — -- / --

| Status | Test | Dev. Inst. Nr. | Obj. Typ | Obj. Inst. Nr. | Property | Beschreibung |
|---|---|---|---|---|---|---|

Bereit

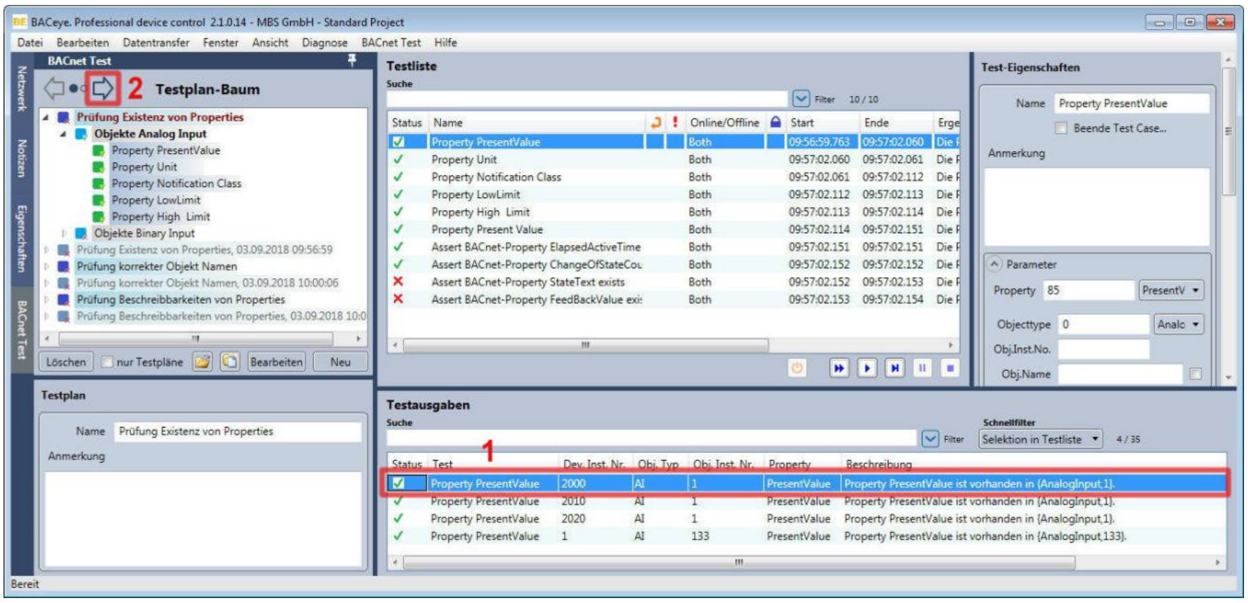
Figure 23

### Navigate to the detailed view for BACnet objects

There are two ways to navigate to the detailed view for a BACnet object (Figure 24):

1. Select the BACnet object entry in the list of test outputs and double-click that row.

2. Select the BACnet object entry in the list of test outputs and click the navigation arrow on the right-hand side in the navigation view ⇦●○⇨.


BACeye. Professional device control 2.1.0.14 - MBS GmbH - Standard Project
Datei Bearbeiten Datentransfer Fenster Ansicht Diagnose BACnet Test Hilfe
Netzwerk
Notizen
Eigenschaften
BACnet Test
2
Testplan-Baum
Löschen
nur Testpläne
Bearbeiten
Neu
1
Bereit

Prüfung Existenz von Properties
- Objekte Analog Input
  - Property PresentValue
  - Property Unit
  - Property Notification Class
  - Property LowLimit
  - Property High Limit
- Objekte Binary Input

Prüfung Existenz von Properties, 03.09.2018 09:56:59
Prüfung korrekter Objekt Namen
Prüfung korrekter Objekt Namen, 03.09.2018 10:00:06
Prüfung Beschreibbarkeiten von Properties
Prüfung Beschreibbarkeiten von Properties, 03.09.2018 10:0

**Testplan**

Name: Prüfung Existenz von Properties
Anmerkung

**Testliste**

Suche — Filter 10 / 10

| Status | Name | Online/Offline | Start | Ende | Erge |
|---|---|---|---|---|---|
| ✓ | Property PresentValue | Both | 09:56:59.763 | 09:57:02.060 | Die F |
| ✓ | Property Unit | Both | 09:57:02.060 | 09:57:02.061 | Die F |
| ✓ | Property Notification Class | Both | 09:57:02.061 | 09:57:02.112 | Die F |
| ✓ | Property LowLimit | Both | 09:57:02.112 | 09:57:02.113 | Die F |
| ✓ | Property High Limit | Both | 09:57:02.113 | 09:57:02.114 | Die F |
| ✓ | Property Present Value | Both | 09:57:02.114 | 09:57:02.151 | Die F |
| ✓ | Assert BACnet-Property ElapsedActiveTime | Both | 09:57:02.151 | 09:57:02.151 | Die F |
| ✓ | Assert BACnet-Property ChangeOfStateCou | Both | 09:57:02.152 | 09:57:02.152 | Die F |
| ✗ | Assert BACnet-Property StateText exists | Both | 09:57:02.152 | 09:57:02.153 | Die F |
| ✗ | Assert BACnet-Property FeedBackValue exi: | Both | 09:57:02.153 | 09:57:02.154 | Die F |

**Test-Eigenschaften**

Name: Property PresentValue
Beende Test Case...
Anmerkung
Parameter
Property: 85 — PresentV
Objecttype: 0 — Analo
Obj.Inst.No.:
Obj.Name:

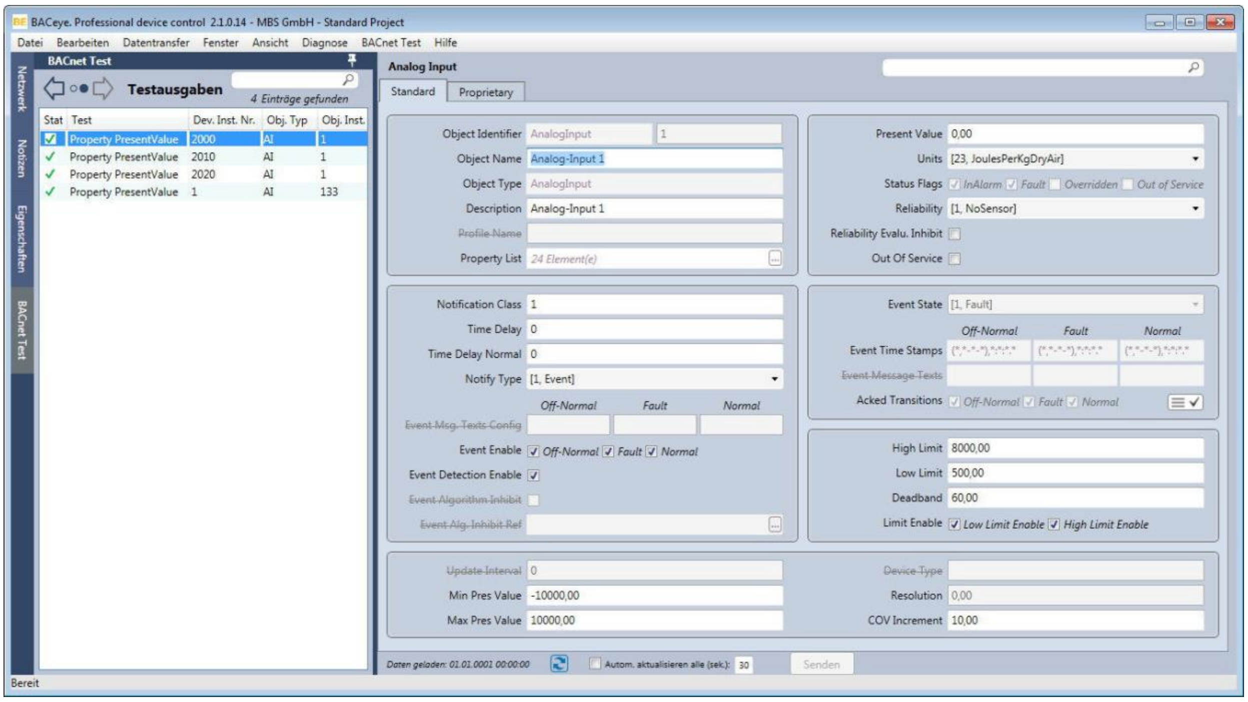
**Testausgaben**

Suche — Filter — Schnellfilter: Selektion in Testliste — 4 / 35

| Status | Test | Dev. Inst. Nr. | Obj. Typ | Obj. Inst. Nr. | Property | Beschreibung |
|---|---|---|---|---|---|---|
| ✓ | Property PresentValue | 2000 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogInput,1}. |
| ✓ | Property PresentValue | 2010 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogInput,1}. |
| ✓ | Property PresentValue | 2020 | AI | 1 | PresentValue | Property PresentValue ist vorhanden in {AnalogInput,1}. |
| ✓ | Property PresentValue | 1 | AI | 133 | PresentValue | Property PresentValue ist vorhanden in {AnalogInput,133}. |

Figure 24

The test outputs list then appears in the BACnet-Test plug-in navigation window. On the right in the detail window you see the detailed view of the referenced BACnet object with detailed properties (Figure 25). To return to the test plan tree, click the left-pointing navigation arrow.


BACeye. Professional device control 2.1.0.14 - MBS GmbH - Standard Project
Datei Bearbeiten Datentransfer Fenster Ansicht Diagnose BACnet Test Hilfe
Netzwerk
Notizen
Eigenschaften
BACnet Test
Standard
Proprietary
Senden
Bereit

**Testausgaben** — 4 Einträge gefunden

| Stat | Test | Dev. Inst. Nr. | Obj. Typ | Obj. Inst. |
|---|---|---|---|---|
| ✓ | Property PresentValue | 2000 | AI | 1 |
| ✓ | Property PresentValue | 2010 | AI | 1 |
| ✓ | Property PresentValue | 2020 | AI | 1 |
| ✓ | Property PresentValue | 1 | AI | 133 |

**Analog Input**

Object Identifier: AnalogInput 1
Object Name: Analog-Input 1
Object Type: AnalogInput
Description: Analog-Input 1
~~Profile Name~~:
Property List: 24 Element(e)

Present Value: 0,00
Units: [23, JoulesPerKgDryAir]
Status Flags: [✓ InAlarm] [✓ Fault] Overridden Out of Service
Reliability: [1, NoSensor]
Reliability Evalu. Inhibit:
Out Of Service:

Notification Class: 1
Time Delay: 0
Time Delay Normal: 0
Notify Type: [1, Event]
~~Event Msg. Texts Config~~: Off-Normal Fault Normal
Event Enable: [✓ Off-Normal] [✓ Fault] [✓ Normal]
Event Detection Enable: [✓]
~~Event Algorithm Inhibit~~:
~~Event Alg. Inhibit Ref~~:

Event State: [1, Fault]
Event Time Stamps: Off-Normal (*,*-*-*),*:*:*.* Fault (*,*-*-*),*:*:*.* Normal (*,*-*-*),*:*:*.*
~~Event Message Texts~~:
Acked Transitions: [✓ Off-Normal] [✓ Fault] [✓ Normal]

High Limit: 8000,00
Low Limit: 500,00
Deadband: 60,00
Limit Enable: [✓ Low Limit Enable] [✓ High Limit Enable]

~~Update Interval~~: 0
Min Pres Value: -10000,00
Max Pres Value: 10000,00

~~Device Type~~:
Resolution: 0,00
COV Increment: 10,00

Daten geladen: 01.01.0001 00:00:00 — Autom. aktualisieren alle (sek.): 30

Figure 25

## Note:

If no BACnet object was referenced beforehand, or if the data is unavailable, a text message on the right in the detail area states that a test output needs to be selected.

# 5 Data storage and transfer

## 5.1 Save and delete test plans and test runs

Every test plan or test run is saved in its own file. The files are saved in a separate directory in the project folder of the current BACeye project (XML format with assigned XSD schema).

Every new test plan or test run is given an ID (GUID) for identification. This ID is used in creating the file name. When a test plan or test run is deleted, the associated file is deleted.

## 5.2 Save and load the BACeye project

When the current BACeye project is saved, the BACnet-Test plug-in creates, in the project backup, a list with the IDs of the existing test plans and test runs in the sequence in which they occur in the test plan tree.

When they are loaded, the test plan tree is rebuilt using this information. When the project is saved, the BACnet-Test plug-in also saves the layout of the columns in the test list and the list of test outputs in the detail and navigation view.

## 5.3 Export and import test plans, test runs and test cases

The BACnet-Test plug-in enables test plans or test runs to be exported and imported to a file. When this is done, test plans (*.tpf) and test runs (*.trf) are exported as XML files. The schema (XSD) used is the same as the one used when saving.

**Export test plans and test runs**

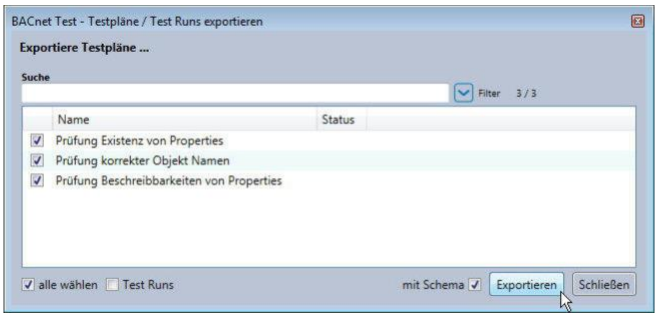
You can export test plans or test runs as follows:

a) Use the context menu for a test plan or test run in the test plan tree
b) Use the button beneath the test plan tree
c) Use the menu path ***BACnet Test > Test plan export…*** (BACnet Test > Testplan-Export…)

If you use the ***Export*** (Exportieren) context menu item in the test plan tree to do the export, the list in the export dialog only contains the selected entry for the context (test plan or test run).

Enter a file name in the Save dialog which then opens, and click **[Save]** (Speichern).

In cases b) and c), a dialog opens with a list of the components that can be selected (Figure 26). Components to be exported can be selected and deselected using the checkbox. Check the *Select all* (alle wählen) checkbox to select all the test plans. Check the *Test runs* box to specify that only test runs are to be selected and exported.


BACnet Test - Testpläne / Test Runs exportieren

**Exportiere Testpläne ...**

Suche

Filter 3 / 3

| | Name | Status |
|---|---|---|
| ✓ | Prüfung Existenz von Properties | |
| ✓ | Prüfung korrekter Objekt Namen | |
| ✓ | Prüfung Beschreibbarkeiten von Properties | |

[✓] alle wählen [ ] Test Runs mit Schema [✓] Exportieren Schließen

Figure 26

The list has search, filter and sort functions. You use the *With schema* (mit Schema) checkbox to specify whether the associated schema file ought also to be exported, or not, when the data is exported.

You start the export with the **[Export]** (Exportieren) button.
Enter a file name in the Save dialog and click **[Save]** (Speichern).
The export's progress is displayed.
A message in a window shows that the export has been successful.

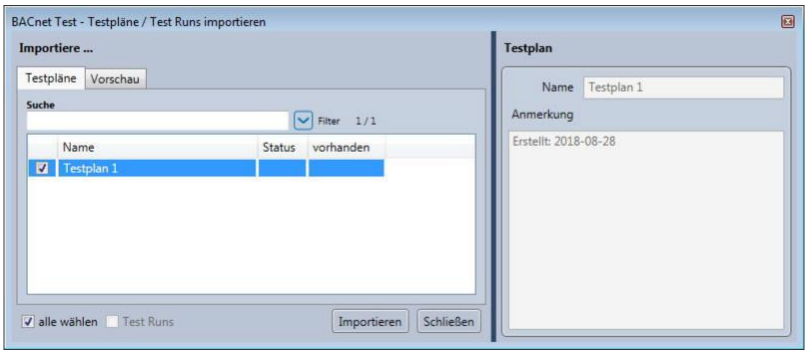
**Import test plans and test runs**

You can import test plans or test runs as follows:

a) Use the button beneath the test plan tree
b) Use the menu path ***BACnet test > Test plan import…*** (BACnet Test > Testplan-Import…)

In both cases, you select the source file in the dialog that opens and click **[Open]** (Öffnen).

The test plans or test runs included are then listed in the import dialog (Figure 27). In the list in the *Test plans* (Testpläne) tab you can select the checkboxes for the test plans or test runs to be imported, and then click **[Import]** (Importieren) to import them.


BACnet Test - Testpläne / Test Runs importieren

Importiere ...

Testpläne | Vorschau

Suche

Filter 1 / 1

| | Name | Status | vorhanden |
|---|---|---|---|
| ✓ | Testplan 1 | | |

✓ alle wählen | Test Runs | Importieren | Schließen

Testplan

Name: Testplan 1

Anmerkung

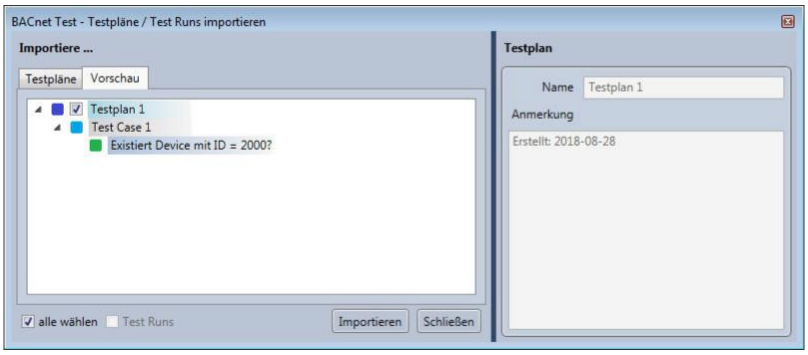
Erstellt: 2018-08-28

Figure 27

The list has search, filter and sort functions. The *Available* (vorhanden) column shows whether the component is available in the system (ticked). The *Test plan* (Testplan) area to the right of the list shows the properties of the component selected as a preview.

The *Preview* (Vorschau) tab shows the test plan tree structure for the component selected (Figure 28).

Start the import with the **[Import]** (Importieren) button.


BACnet Test - Testpläne / Test Runs importieren

Importiere ...

Testpläne | Vorschau

- ✓ Testplan 1
  - Test Case 1
    - Existiert Device mit ID = 2000?

✓ alle wählen | Test Runs | Importieren | Schließen

Testplan

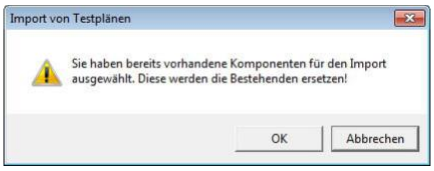
Name: Testplan 1

Anmerkung

Erstellt: 2018-08-28

Figure 28

**Note:**
If you import any components that already exist, a confirmation dialog will notify you that the existing components will be replaced (Figure 29).


Import von Testplänen

Sie haben bereits vorhandene Komponenten für den Import ausgewählt. Diese werden die Bestehenden ersetzen!

Figure 29

**Export and import test cases**

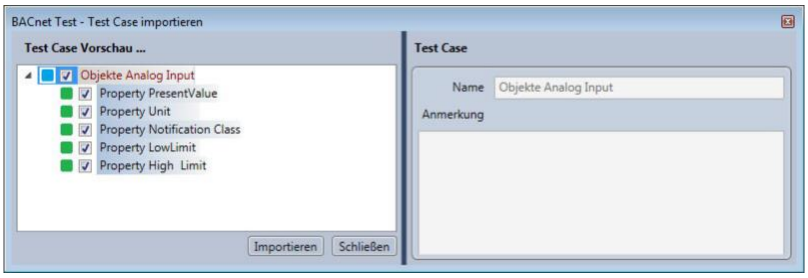
You export test cases using the context menu for a test case in the test plan tree.

- Click the ***Export*** (Exportieren) context menu item, enter a file name (*.tcf) in the Save dialog that opens, and click **[Save]** (Speichern).

You import test cases using the context menu for a test plan in the test plan tree.

- Click the ***Import test case*** (Test Case importieren) context menu item, select the source file in the Save dialog that opens, and click **[Open]** (Öffnen).

The test case it contains is then shown as a preview in the import dialog (Figure 30).


BACnet Test - Test Case importieren

Test Case Vorschau ...

- ✓ Objekte Analog Input
  - ✓ Property PresentValue
  - ✓ Property Unit
  - ✓ Property Notification Class
  - ✓ Property LowLimit
  - ✓ Property High Limit

Test Case

Name: Objekte Analog Input

Anmerkung

Figure 30

In the list, the components to be imported – e.g. individual tests – are selected and deselected using the checkboxes.

- Start the import with the **[Import]** (Importieren) button.