

a:control
OPC-UA



Diagraph GmbH
Friedrich-Bergius-Ring 30
97076 Würzburg
Germany

Phone: +49 (0) 931-25076-0
FAX: +49 (0) 931 25076-50

E-Mail: support@diagraph.de
Web: www.diagraph.de

Table of Contents

1. SYSTEM REQUIREMENTS	3
2. FOREWORD	4
3. THE PROTOCOL	4
4. OPC UA – INDUSTRY 4.0	4
5. OPC UA VIA THE A:CONTROL SOFTWARE WITH PRINTERS FROM DIAGRAPH / ALLEN CODING	4
6. CHARACTERISTICS	5
7. PROGRAM VERSIONS	7
8. INSTALLATION IN A:CONTROL / LICENSING OF THE WEB SERVICES	77
9. TESTING OPC ACCESS	8
10. TEST CASES	9
TEST CASE 1: LOADING A PRINT FORMAT AND CHANGING A VARIABLE	9
TEST CASE 2: ERROR QUERY AND LOAD ERROR DETAILS	11
11. OPC UA CLIENT SAMPLE (MICROSOFT C# .NET)	16
12. OPC-UA VIA ACONTROL-WEBGUI AND NODE-RED	17
13. PROPERTIES OF THE WEB BOX	19
14. WEBBOX OPC DEMO WITH NODE-RED	21

Note:

All screenshots and pictures of the application which are shown in this document are to be taken as examples and are subjected to the copyright of Diagraph GmbH.

1. System Requirements

PC hardware	requirements
Installed a:control from version aControl-1.12-R, a:Control WebGUI	CPU with at least 2.0 GHz processor speed
Physical main memory	Minimum required to complete the installation: 4 GB Recommendation: 8 GB
Free disk space	150 MB for program installation + storage space for print formats
Mouse, keyboard, monitor or touch screen	Resolution monitor at least 800x600, 16 bit color depth
Network card	Ethernet
RS232 / COM ports	One COM port for each serially connected printer
USB ports	USB A for data exchange with USB stick
software	Windows 7 or higher Microsoft .NET Framework 4.8

It is quite possible that the software will run perfectly on different systems, but the program performance could suffer with weaker hardware. Above all, sufficient free memory + hard disk space are an important factor for problem-free program execution.

printer hardware	requirements
Printer type	All printers supported in a:control with a:Control WebGUI
Firmware	Latest printer firmware
Connectivity	Ethernet network

2. Foreword

In industry, more and more production and logistics environments are based on digital process control systems. Machines exchange information with people or with other machines, process production-relevant data autonomously, enter data or forward it. Customization, scalability, lean and traceable processes as well as reliable devices and components are important requirements.

Most of the applications or services on which the control systems are based are hosted on the Internet. Data from sensors, devices or software is transferred to the cloud and can easily be transferred between printers or labeling systems, robots and IT systems are exchanged - provided they all speak the same language and understand each other.

3. The protocol

Open Platform Communications Unified Architecture, abbreviation OPC UA creates the basis for the interaction of machines and components from different manufacturers in an industrial environment. In OPC UA, various specifications have been combined into a consistent data model, creating a new interface standard. OPC UA triggers networking in industrial production, as we know it from network and USB interfaces in offices.

4. OPC UA – Industry 4.0

Connected to Diagraph / Allen Coding printers

If an IP-enabled network is available, OPC UA can already be used as a consistent framework for accessing information from machine to machine.

When marking components, products or packaging within a plant, the current generation of Allen Coding printers and Diagraph inkjet printers are already OPC UA-capable.

They can work together with higher-level control units.

OPC UA highlights at a glance

- On Demand Production
- Automated processes
- device configuration
- Remote control of the device
- device monitoring
- interoperability
- prognosis and analysis

OPC UA as a superior control system

Labels can be loaded remotely, label properties read and variable data exchanged for printing.

OPC UA as a superior control system offers the advantage of bringing together production data, alarms, events and data history controlled by software under one roof.

It integrates and unifies different address spaces and the interfaces to be accessed.

5. OPC UA via the a:control software with printers from Diagraph / Allen Coding

The following OPC UA functions are currently supported by a:Control for Diagraph / Allen Coding printers:

communication stack
OPC UA binary protocol
encryption
no
Basic128Rsa15
Basic256Sha256
authentication
anonymous
Username Password

While fully authenticated users may have full access to the data model provided, Anonymous users can only access printer nodes in read-only mode and are not allowed to invoke any method, even if the method does not change the printer's configuration or state.

OPC UA services

Secure Channel Service Set
Session Service Set
Attribute service set: read/write attributes, events, no access to historical data
MethodServiceSet
Monitored Item Service Set
Subscription Service Set
Show service set

namespaces

Two namespaces are available:

opc.tcp://<MyIP>:62557/ITWMC/OPCServer to monitor and control print jobs

OPC Watch

TRAEGER.de

Server Browser

Configuration Monitor Analyse

Read Write Auto Update Copy Node ID (1) opc.tcp://192.168.173.55:62557/ITWMC/OPCServer/Objects/Printer

Name	Value	Type	Base Type	Status	Server Timestamp	Source Timestamp	AccessLevel	Description	NodeId
GetConnectionStatus		Method			06.03.2023 19:57:16		None		ns=2;i=14
GetErrorDetails		Method			06.03.2023 19:57:16		None		ns=2;i=32
GetErrorStatus		Method			06.03.2023 19:57:16		None		ns=2;i=29
GetFontNames		Method			06.03.2023 19:57:16		None		ns=2;i=41
GetFormat		Method			06.03.2023 19:57:16		None		ns=2;i=44
GetFormatNames		Method			06.03.2023 19:57:16		None		ns=2;i=38
GetNativeDeviceInfo		Method			06.03.2023 19:57:16		None		ns=2;i=109
GetNativeDeviceInfoAttributes		Method			06.03.2023 19:57:16		None		ns=2;i=105
GetNativeDeviceInfoNames		Method			06.03.2023 19:57:16		None		ns=2;i=113
GetNativeProperty		Method			06.03.2023 19:57:16		None		ns=2;i=97
GetNativePropertyAttributes		Method			06.03.2023 19:57:16		None		ns=2;i=93
GetNativePropertyNames		Method			06.03.2023 19:57:16		None		ns=2;i=90
GetPrintCounter		Method			06.03.2023 19:57:16		None		ns=2;i=35
GetPrinterInfo		Method			06.03.2023 19:57:16		None		ns=2;i=4
GetProperty		Method			06.03.2023 19:57:16		None		ns=2;i=82
GetPropertyAttribute		Method			06.03.2023 19:57:16		None		ns=2;i=78
GetPropertyNames		Method			06.03.2023 19:57:16		None		ns=2;i=75
GetVariable		Method			06.03.2023 19:57:16		None		ns=2;i=67
GetVariableNames		Method			06.03.2023 19:57:16		None		ns=2;i=64
Lock		Method			06.03.2023 19:57:16		None		ns=2;i=7
PrintFormat		Method			06.03.2023 19:57:16		None		ns=2;i=60
Reboot		Method			06.03.2023 19:57:16		None		ns=2;i=23
RenameFormat		Method			06.03.2023 19:57:16		None		ns=2;i=52
ResetError		Method			06.03.2023 19:57:16		None		ns=2;i=20
SendFormat		Method			06.03.2023 19:57:16		None		ns=2;i=56
SetFormat		Method			06.03.2023 19:57:16		None		ns=2;i=48
SetNativeProperty		Method			06.03.2023 19:57:16		None		ns=2;i=101
SetProperty		Method			06.03.2023 19:57:16		None		ns=2;i=86
SetVariable		Method			06.03.2023 19:57:16		None		ns=2;i=71
Stop		Method			06.03.2023 19:57:16		None		ns=2;i=26
TriggerPrint		Method			06.03.2023 19:57:16		None		ns=2;i=17
Unlock		Method			06.03.2023 19:57:16		None		ns=2;i=11
Version	"1.12.15703-RC"	String		Good - The ope	06.03.2023 19:57:16	06.03.2023 19:57:16	CurrentRead		ns=2;i=2

Connected!

6. Characteristics

Knotenname	Art	ID	Description	JSON-Serialized return values
GetConnectionStatus	Method	ns=2;i=14	Printer connection status	bool
GetErrorDetails	Method	ns=2;i=32	Reading the error details	List<string>
GetErrorStatus	Method	ns=2;i=29	Reading the error status	string
GetFontNames	Method	ns=2;i=41	Installed fonts	List<string>
GetFormat	Method	ns=2;i=44	Read native printer format	string
GetFormatNames	Method	ns=2;i=38	List of existing print images	List<string>
GetNativeDeviceInfo	Method	ns=2;i=109	Read native device info	string
GetNativeDeviceInfoAttributes	Method	ns=2;i=105	List of native device info attributes	List<string>
GetNativeDeviceInfoNames	Method	ns=2;i=113	Name list of the device information	List<string>
GetNativeProperty	Method	ns=2;i=97	Read native property	string
GetNativePropertyAttributes	Method	ns=2;i=93	List of native property attributes	List<string>
GetNativePropertyNames	Method	ns=2;i=90	Name list of properties	List<string>
GetPrintCounter	Method	ns=2;i=35	number of prints	long
GetPrinterInfo	Method	ns=2;i=4	General printer information	List<KeyValuePair<string, string>>
GetProperty	Method	ns=2;i=82	Read printer property	string
GetPropertyAttribute	Method	ns=2;i=78	Read printer property attributes	string
GetPropertyNames	Method	ns=2;i=75	List of printer properties	List<string>
GetVariable	Method	ns=2;i=67	Read variable content	string
GetVariableNames	Method	ns=2;i=64	Variable names of the current print format	List<string>
Lock	Method	ns=2;i=7	UI Lock	bool
PrintFormat	Method	ns=2;i=60	Load print image	string

Reboot	Method	ns=2;i=23	Restart printer	bool
RenameFormat	Method	ns=2;i=52	Rename print image	bool
ResetError	Method	ns=2;i=20	reset errors	bool
SendFormat	Method	ns=2;i=56	Send native binary print format to printer (Base64)	string
SetFormat	Method	ns=2;i=48	Set print format as active	bool
SetNativeProperty	Method	ns=2;i=101	Send native sequence to printer	string
SetProperty	Method	ns=2;i=86	write property	bool
SetVariable	Method	ns=2;i=71	Write variable content	bool
Stop	Method	ns=2;i=26	to stop the printer	bool
TriggerPrint	Method	ns=2;i=17	trigger pressure	bool
Unlock	Method	ns=2;i=11	Unlock UI	bool
Version	String	ns=2;i=2	version	bool

7. Program versions

The Diagraph OPC-UA interface represents a link between OPC-UA and Allen/Diagraph printers, making them OPC-capable.

The OPC UA interface is based on the REST services of a running a:control instance. As a prerequisite, a license for web services must be installed in a:control. aControl-WebGui has its license on the SD card that is in the web box.

The responses of complex OPC methods are provided in JSON format.

8. Installation in a:control / licensing of the web services

The OPC-UA interface from a:control uses the web/REST service functionality internally. So it has to be licensed.

One Client Access License (CAL) is required for each printer set up.

- **Settings in a:control**

In order to be able to use the OPC UA functionality, a:control must be started as an administrator and in the tab **Program Settings->Configuration->Web Service** to be activated.

The screenshot shows the 'Settings' window with the 'SERVER' tab selected. The 'WEB SERVICE' sub-tab is active. At the top right, there are 'Inactive' and 'Active' buttons, with 'Active' being highlighted in orange. Below this, the URL 'http://192.168.175.70:8086/swagger' is displayed. The 'WS URL' field contains 'http://192.168.175.70:8086/acontrol/webservice', the 'REST' field contains 'http://192.168.175.70:8086/acontrol', and the 'OPC-UA' checkbox is checked. The 'OPC-UA' field contains 'opc.tcp://192.168.175.70:62557/ITWMC/OPCServer'. The 'User' field contains 'diagraph'. The 'Password' field is empty. There are 'API' and 'Firewall' buttons. At the bottom, there are 'OK', 'Cancel', and 'Apply' buttons.

If the OPC UA option is activated, the OPC address

opc.tcp://<myIP>:62557/ITWMC/OPCServer

opc.tcp://127.0.0.1:62557/ITWMC/OPCServer can accessed from the local PC.

9. Testing OPC access

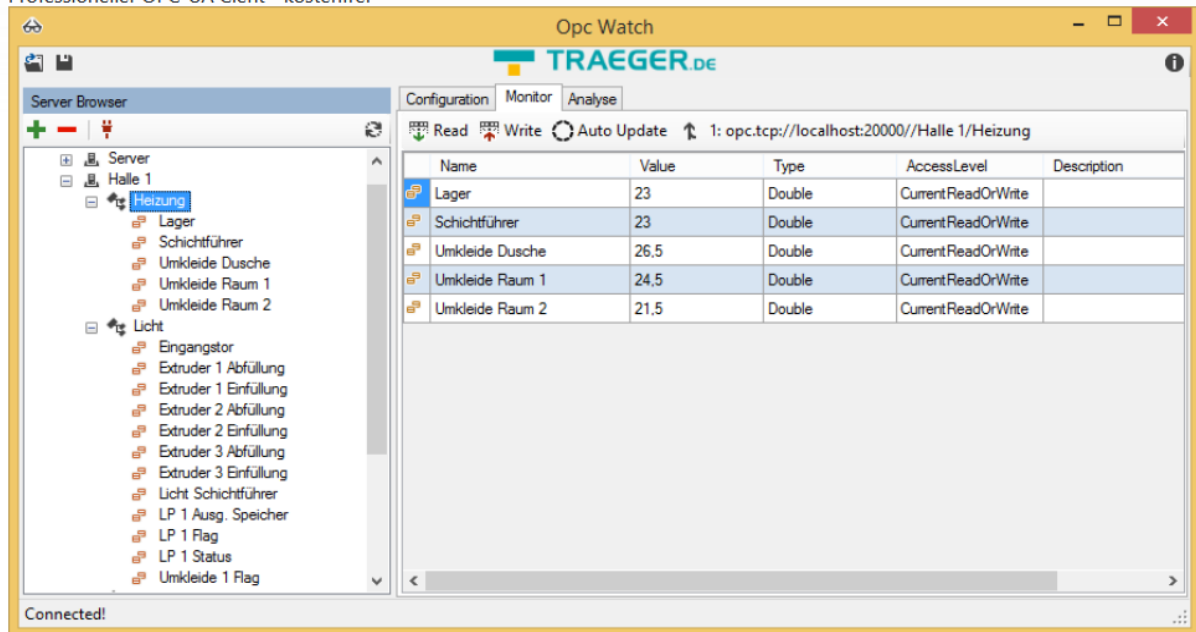
To test the OPC UA functionality, you can, for example, from the website www.traeger.de the free OPC Watch program can be loaded:

<https://www.traeger.de/downloads/opc-ua-development-toolkit>

OPC Watch - kostenfreier OPC UA Client

Version 2019-09-25-2.8.1.0

Professioneller OPC-UA Client - kostenfrei



[Download](#)

[Details](#)

After the OPC UA client has been installed, the program can be started.

After entering the address `opc.tcp://127.0.0.1:62557/ITWMC/` the methods and properties provided can be tested with this client.

10. Test cases

Test case 1: Loading a print format and changing a variable

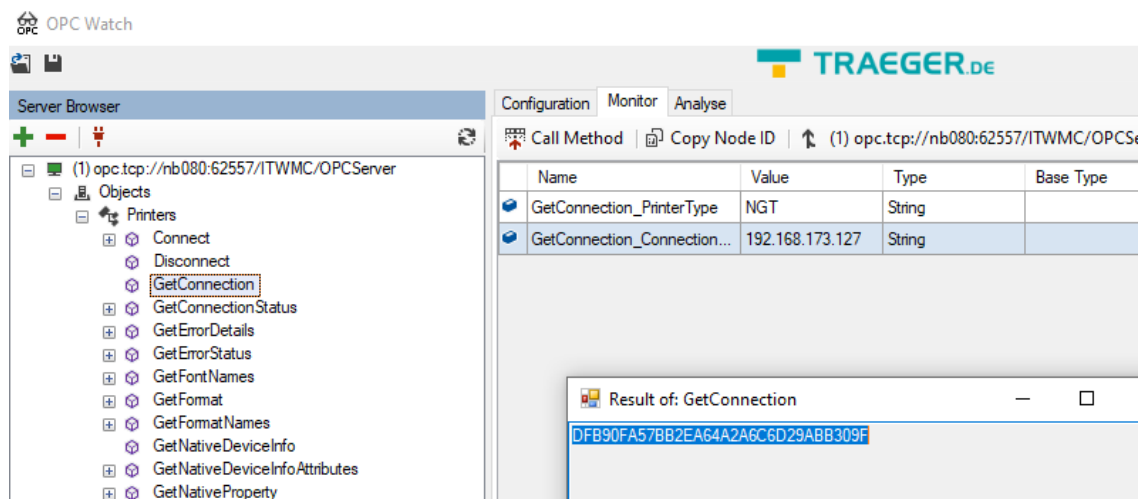
All printers created in a:control have a PrinterID. Printers that have already been created can be read using the "GetPrinterIDs" method. If this is not known, or if only the printer type and the printer IP are known, the PrinterID can be determined using the "GetConnection" method:

Printer Types:

MLI

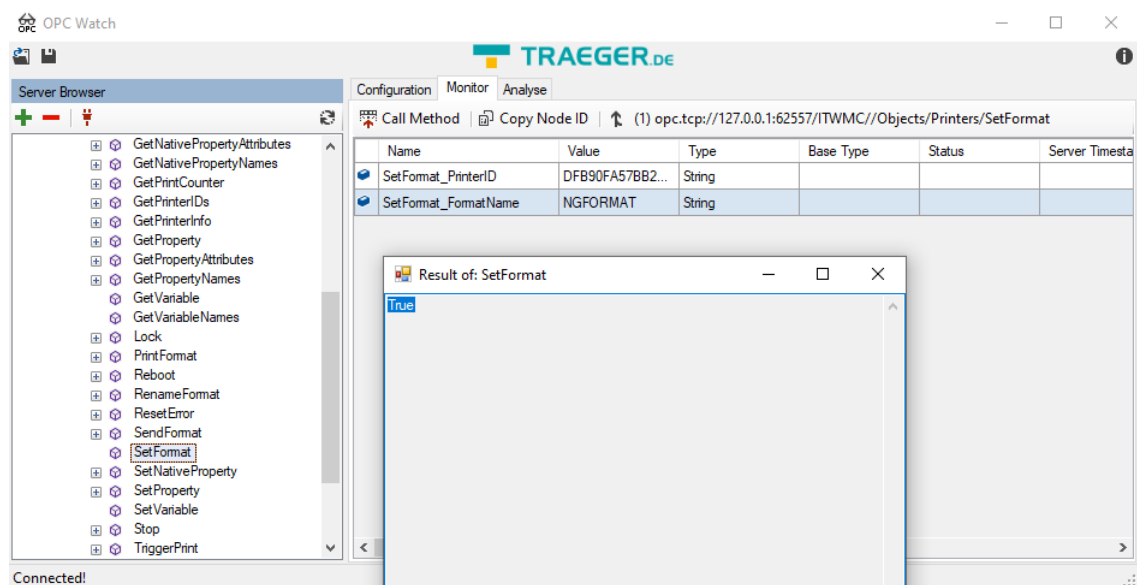
NGT

XL

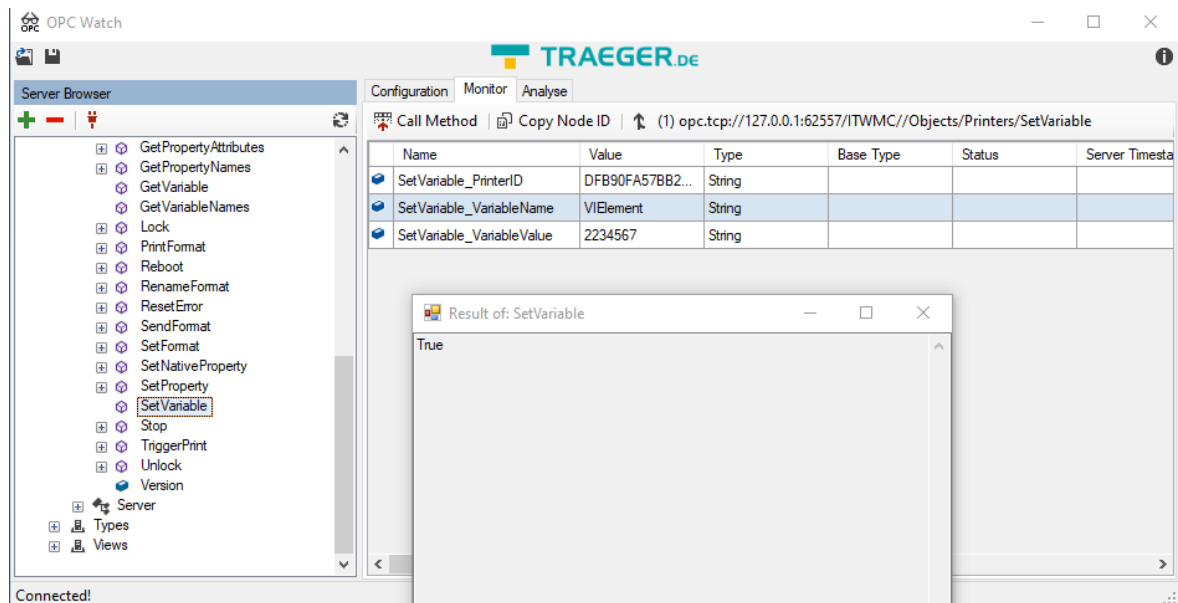


DFB90FA57BB2EA64A2A6C6D29ABB309F

As soon as the connection is established, a print image can be loaded with the ID:



Then the variable can then be set:



After completing all tasks, the printer connection should be explicitly released again using the "Disconnect" method:

The screenshot shows the OPC Watch application window. The title bar reads 'OPC Watch'. The main interface has a 'Server Browser' on the left, a 'Configuration' tab, and a 'Monitor' tab. The 'Monitor' tab is active, showing a table with columns: Name, Value, Type, Base Type, and Status. The table contains one entry: 'Disconnect_PrinterID' with a value of 'A57BB2EA64A2A6C6D29ABB309F' and a type of 'String'. A dialog box titled 'Result of: Disconnect' is open in the foreground, displaying the value 'True'. The 'Server Browser' on the left shows a tree structure under 'Printers' with various methods listed, including 'Connect', 'Disconnect', 'GetConnectionStatus', 'GetErrorDetails', 'GetErrorStatus', 'GetFontNames', 'GetFormat', 'GetFormatNames', 'GetNativeDeviceInfo', 'GetNativeDeviceInfoAttributes', 'GetNativeProperty', 'GetNativePropertyAttributes', 'GetNativePropertyNames', 'GetPrintCounter', 'GetPrinterIDs', 'GetPrinterInfo', 'GetProperty', 'GetPropertyAttributes', 'GetPropertyNames', and 'GetVariable'. The 'Disconnect' method is highlighted. The status bar at the bottom left indicates 'Connected!'.

Name	Value	Type	Base Type	Status
Disconnect_PrinterID	A57BB2EA64A2A6C6D29ABB309F	String		

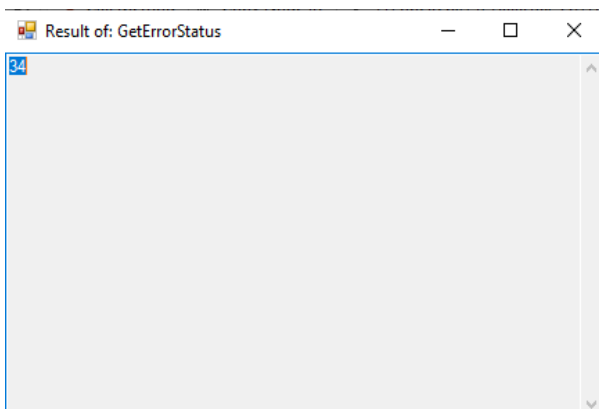
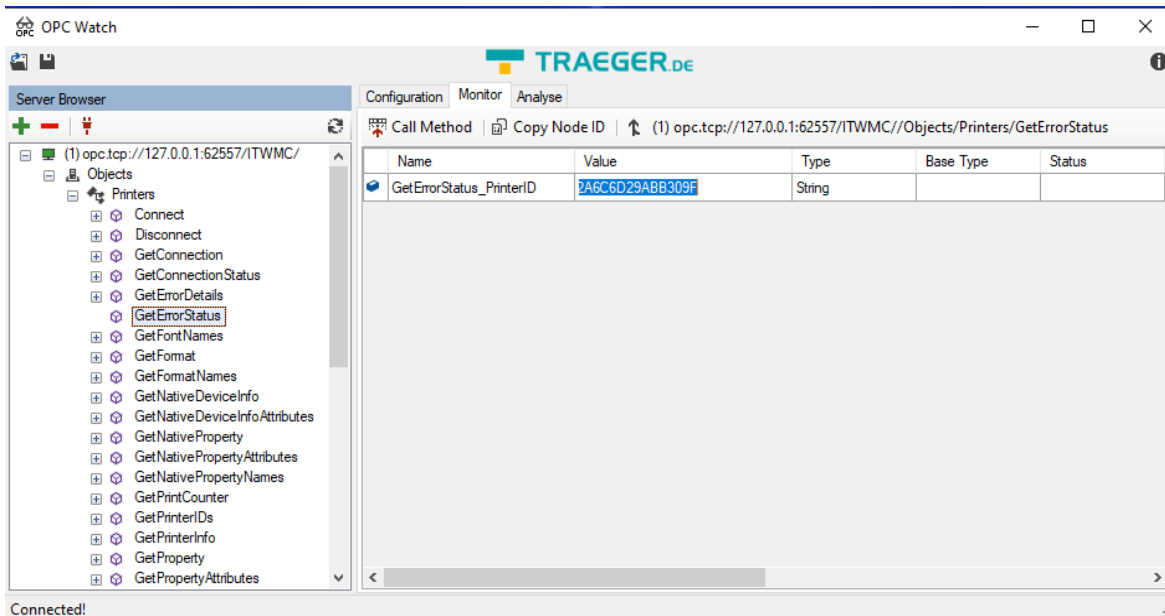
Result of: Disconnect

True

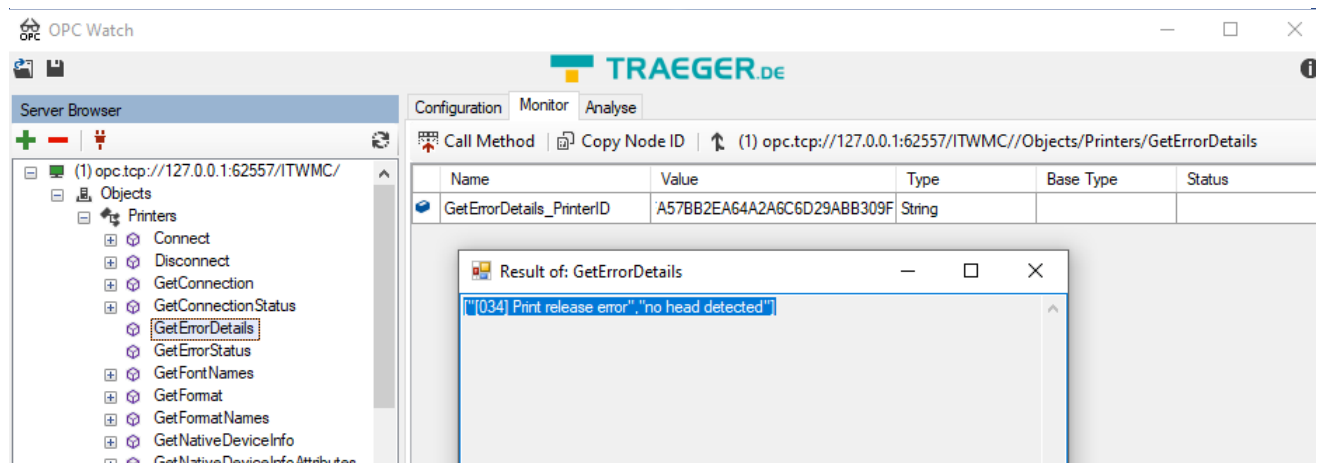
Connected!

Test case 2: Error Query and load error details

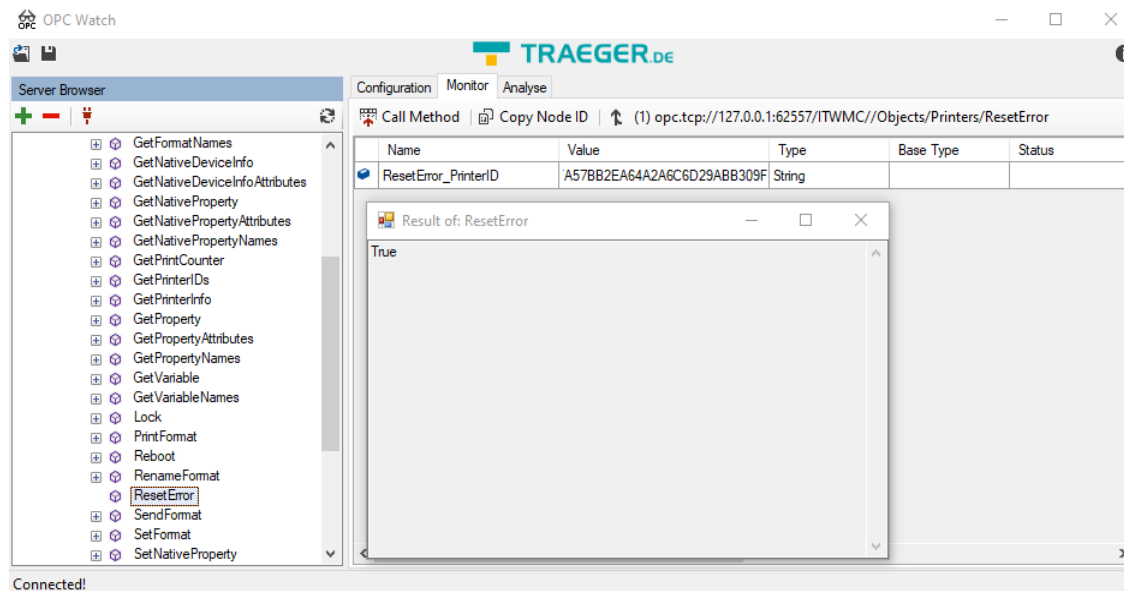
First, a connection to the printer must be established (see test case 1). If the PrinterID is already known, the printer can be queried immediately for errors.



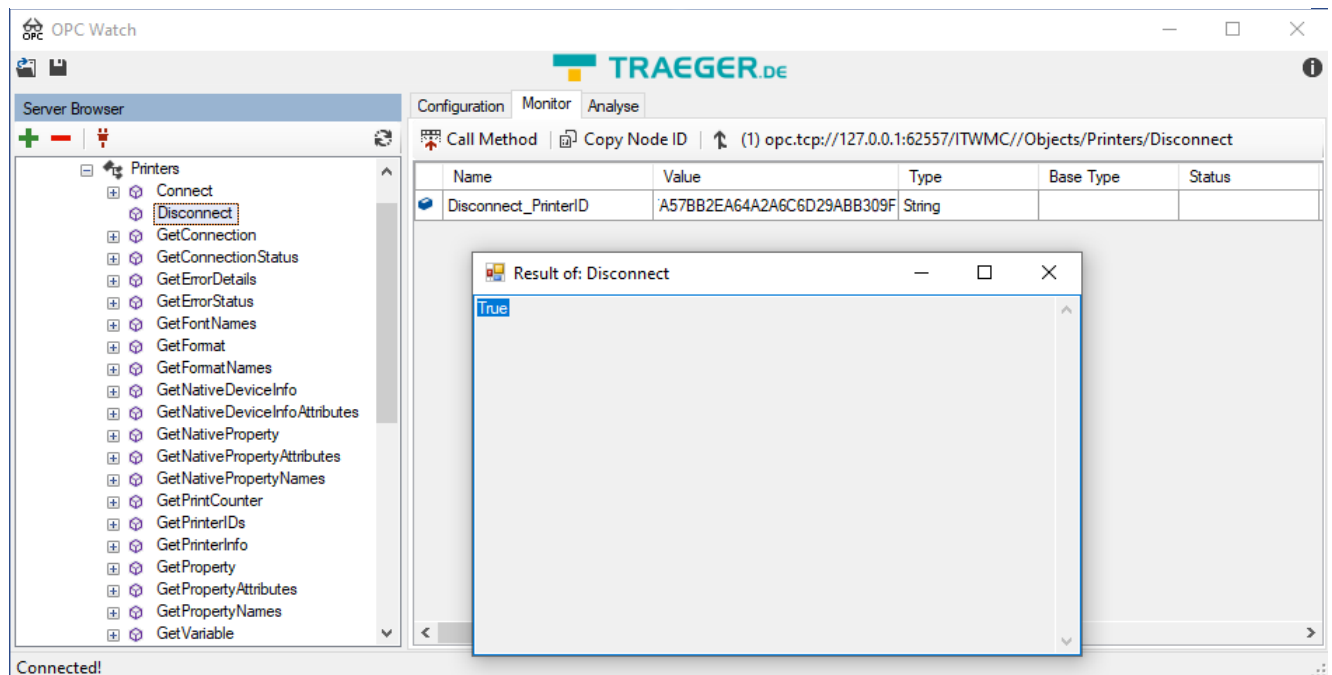
The "GetErrorDetails" method provides further information about the pending printer error:



The pending error can now be acknowledged with the "Reset Error" method:

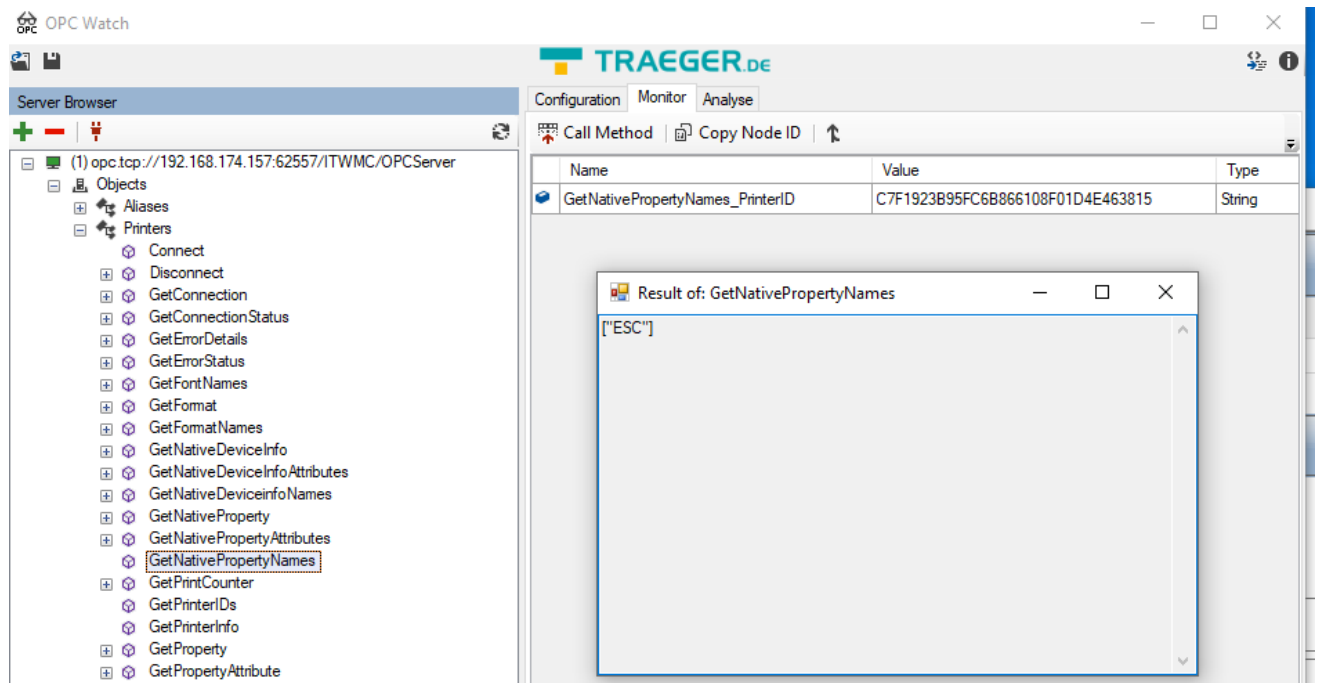


After completing all tasks, the printer connection should be explicitly released again using the "Disconnect" method:



Test Case 3: Query native Esc sequences on NGT printers

First, a connection to the printer must be established (see test case 1). When connected to the printer, can the Esc sequence is triggered and the result is queried. There is a property for an NGT printer



OPC Watch

Server Browser

(1) opc.tcp://192.168.174.157:62557/ITWMC/OPCServer

Objects

Aliases

Printers

Connect

Disconnect

GetConnection

GetConnectionStatus

GetErrorDetails

GetErrorStatus

GetFontNames

GetFormat

GetFormatNames

GetNativeDeviceInfo

GetNativeDeviceInfoAttributes

GetNativeDeviceInfoNames

GetNativeProperty

GetNativePropertyAttributes

GetNativePropertyNames

GetPrintCounter

GetPrinterIDs

GetPrinterInfo

GetProperty

GetPropertyAttribute

GetPropertyNames

GetVariable

GetVariableNames

Lock

PrintFormat

Reboot

RenameFormat

ResetError

SendFormat

SetFormat

SetNativeProperty

SetProperty

SetVariable

Stop

TriggerPrint

Unlock

Version

Server

Types

Views

TRAEEGER.DE

Configuration

Monitor

Analyse

Call Method

Copy Node ID

Name	Value	Type
SetNativeProperty_PrinterID	C7F1923B95FC6B866108F01D4E463815	String
SetNativeProperty_NativePropertyName	ESC	String
SetNativeProperty_NativePropertyValue	R20	String

Connected!

Diagraph GmbH
Friedrich-Bergius-Ring 30
D-97076 Würzburg

Tel. +49 931 250 76-0
Fax +49 931 250 76-50

support@diagraph.de
www.diagraph.de

Page 18 / 27
Version: 2026.1
Date: 03/25/2026

The screenshot shows the OPC Watch application window. The left pane displays a tree view of the OPC UA object model. The right pane shows the 'Call Method' dialog with a table of parameters.

Server Browser Tree View:

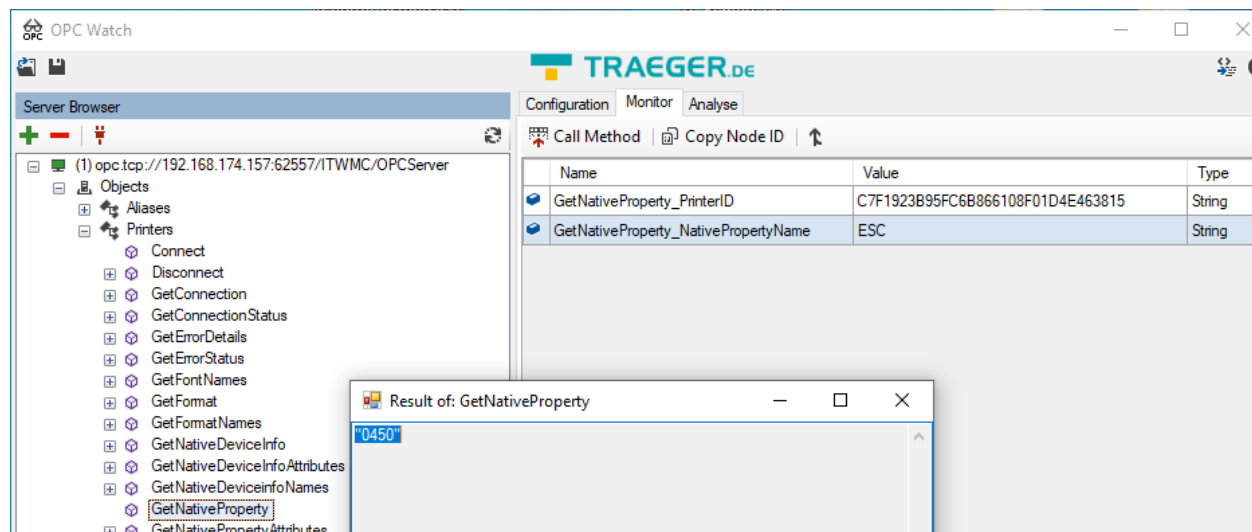
- (1) opc.tcp://192.168.174.157:62557/ITWMC/OPCServer
 - Objects
 - Aliases
 - Printers
 - Connect
 - Disconnect
 - GetConnection
 - GetConnectionStatus
 - GetErrorDetails
 - GetErrorStatus
 - GetFontNames
 - GetFormat
 - GetFormatNames
 - GetNativeDeviceInfo
 - GetNativeDeviceInfoAttributes
 - GetNativeDeviceInfoNames
 - GetNativeProperty
 - GetNativePropertyAttributes
 - GetNativePropertyNames
 - GetPrintCounter
 - GetPrinterIDs
 - GetPrinterInfo
 - GetProperty
 - GetPropertyAttribute
 - GetPropertyNames
 - GetVariable
 - GetVariableNames
 - Lock
 - PrintFormat
 - Reboot
 - RenameFormat
 - ResetError
 - SendFormat
 - SetFormat
 - SetNativeProperty**
 - SetProperty
 - SetVariable
 - Stop
 - TriggerPrint
 - Unlock
 - Version
 - Server
 - Types
 - Views

Call Method Dialog Table:

Name	Value	Type
SetNativeProperty_PrinterID	C7F1923B95FC6B866108F01D4E463815	String
SetNativeProperty_NativePropertyName	ESC	String
SetNativeProperty_NativePropertyValue	R20	String

Connected!

The read value can be queried with



11. OPC UA Client sample (Microsoft C# .NET)

An example of how error polling is carried out via OPC-UA is available as source code.

In this example, a serially connected printer with the name "NGT serial" is polled for errors via the OPC interface in a:control.

The OPC server address in this example is **opc.tcp://127.0.0.1:62557/ITWMC/OPCServer** and can be changed in the setting **ServerURL**

Downloads:

<https://www.diagraph.de/wp-content/uploads/software/ngt/NGT-OPCTest.zip>

```
8 - Running...Press Ctrl-C to exit...
ServerStatusCurrentTime: 27.02.2020 14:53:30, 27.02.2020 14:53:30, Good
PrinterID: 36670853EDFFE42B3112366FC362A1F: Error: [034] Druckfreigabefehler
PrinterID: 36670853EDFFE42B3112366FC362A1F: Error: no head detected
ServerStatusCurrentTime: 27.02.2020 14:53:32, 27.02.2020 14:53:32, Good
PrinterID: 36670853EDFFE42B3112366FC362A1F: Error: [034] Druckfreigabefehler
PrinterID: 36670853EDFFE42B3112366FC362A1F: Error: no head detected
ServerStatusCurrentTime: 27.02.2020 14:53:37, 27.02.2020 14:53:37, Good
```

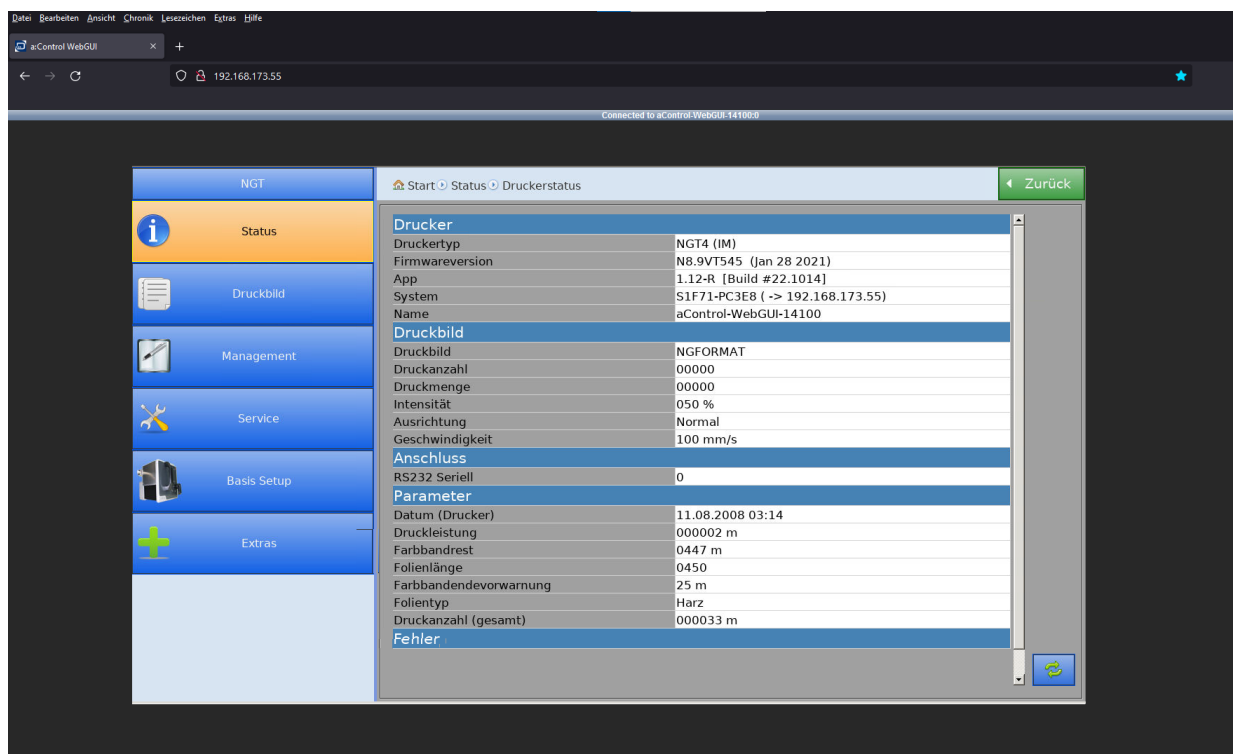
12. OPC-UA via aControl-WebGUI and Node-Red



The aControl-WebGUI is an extension box for Allen/Diagraph printers that offers the printers an a:Control web interface.

A printer is serially connected to the extension box (WebBox) and mirrors the display in the web browser.

By default, the WebBox is accessible at the address <http://192.168.173.55/>



REST Services

The WebBox provides both rest services and OPC UA functionality for the connected printer.

The rest services can be accessed directly from the URL using a Firefox browser

<http://192.168.173.55:8086/acontrol/help>

There are further instructions for the web/rest services available in a separate document.

```

0: "http://192.168.173.55:8086/acontrol/Version"
1: "http://192.168.173.55:8086/acontrol/Logfile"
2: "http://192.168.173.55:8086/acontrol/RebootUI"
3: "http://192.168.173.55:8086/acontrol/GetPrinterIDs"
4: "_"
5: "PrinterName: NGT"
6: "PrinterID: [B6D1217C326B6765B270EF22E2F823AA, NGT]"
7: "PrinterType: NGT"
8: "Interface: COM1,115200,2,8,1,2"
9: "ConnectionID: COET00B30CYJJC1IX6QWNRZ"
10: "_"
▼ 11: "http://192.168.173.55:8086/acontrol/Lock/B6D1217C326B6765B270EF22E2F823AA/True"
▼ 12: "http://192.168.173.55:8086/acontrol/Lock/B6D1217C326B6765B270EF22E2F823AA/False"
▼ 13: "http://192.168.173.55:8086/acontrol/UnLock/B6D1217C326B6765B270EF22E2F823AA"
▼ 14: "http://192.168.173.55:8086/acontrol/GetPrinterInfo/B6D1217C326B6765B270EF22E2F823AA"
▼ 15: "http://192.168.173.55:8086/acontrol/GetConnectionStatus/B6D1217C326B6765B270EF22E2F823AA"
▼ 16: "http://192.168.173.55:8086/acontrol/TriggerPrint/B6D1217C326B6765B270EF22E2F823AA"
▼ 17: "http://192.168.173.55:8086/acontrol/ResetError/B6D1217C326B6765B270EF22E2F823AA"
▼ 18: "http://192.168.173.55:8086/acontrol/Reboot/B6D1217C326B6765B270EF22E2F823AA"
▼ 19: "http://192.168.173.55:8086/acontrol/Stop/B6D1217C326B6765B270EF22E2F823AA"
▼ 20: "http://192.168.173.55:8086/acontrol/GetErrorStatus/B6D1217C326B6765B270EF22E2F823AA"
▼ 21: "http://192.168.173.55:8086/acontrol/GetErrorDetails/B6D1217C326B6765B270EF22E2F823AA"
▼ 22: "http://192.168.173.55:8086/acontrol/GetPrintCounter/B6D1217C326B6765B270EF22E2F823AA"
▼ 23: "http://192.168.173.55:8086/acontrol/GetFormatNames/B6D1217C326B6765B270EF22E2F823AA"
▼ 24: "http://192.168.173.55:8086/acontrol/GetFontNames/B6D1217C326B6765B270EF22E2F823AA"
▼ 25: "http://192.168.173.55:8086/acontrol/GetPrinterLogfile/B6D1217C326B6765B270EF22E2F823AA"
▼ 26: "http://192.168.173.55:8086/acontrol/GetFormat/B6D1217C326B6765B270EF22E2F823AA/{formatName}"
▼ 27: "http://192.168.173.55:8086/acontrol/SetFormat/B6D1217C326B6765B270EF22E2F823AA/{formatName}"
▼ 28: "http://192.168.173.55:8086/acontrol/RenameFormat/B6D1217C326B6765B270EF22E2F823AA/{oldFormatName}/{newFormatName}"
▼ 29: "http://192.168.173.55:8086/acontrol/PrintFormat/B6D1217C326B6765B270EF22E2F823AA/{formatName}"
▼ 30: "http://192.168.173.55:8086/acontrol/SendFormat/B6D1217C326B6765B270EF22E2F823AA/{formatName} POST{stream content}"
▼ 31: "http://192.168.173.55:8086/acontrol/GetPropertyNames/B6D1217C326B6765B270EF22E2F823AA"
▼ 32: "http://192.168.173.55:8086/acontrol/GetPropertyAttribute/{propertyName}"
▼ 33: "http://192.168.173.55:8086/acontrol/GetProperty/B6D1217C326B6765B270EF22E2F823AA/{propertyName}"
▼ 34: "http://192.168.173.55:8086/acontrol/GetAllProperties/B6D1217C326B6765B270EF22E2F823AA"
▼ 35: "http://192.168.173.55:8086/acontrol/SetProperty/B6D1217C326B6765B270EF22E2F823AA/{propertyName}/{propertyValue}"
▼ 36: "http://192.168.173.55:8086/acontrol/SetPropertyValue/B6D1217C326B6765B270EF22E2F823AA/{propertyName} POST{stream content}"
▼ 37: "http://192.168.173.55:8086/acontrol/GetVariableNames/B6D1217C326B6765B270EF22E2F823AA"
▼ 38: "http://192.168.173.55:8086/acontrol/GetVariable/B6D1217C326B6765B270EF22E2F823AA/{variableName}"
▼ 39: "http://192.168.173.55:8086/acontrol/SetVariable/B6D1217C326B6765B270EF22E2F823AA/{variableName}/{variableValue}"
▼ 40: "http://192.168.173.55:8086/acontrol/SetVariableValue/B6D1217C326B6765B270EF22E2F823AA/{variableName} POST{stream content}"
41: "_"

```

OPC UA functions of the WebBox

The OPC UA functionality can be reached via the following address:

opc.tcp://192.168.173.55:62557/ITWMC/OPCServer

The OPC Watch software shows the following functions:

The screenshot shows the OPC Watch software interface. On the left, a tree view shows the object hierarchy: (1) opc.tcp://192.168.173.55:62557/ITWMC/OPCServer/Objects/Printer. The main window displays a table of functions for the selected object. The table has columns: Name, Value, Type, Base Type, Status, Server Timestamp, Source Timestamp, and AccessLevel. The functions listed include GetConnectionStatus, GetErrorDetails, GetErrorStatus, GetFontNames, GetFormat, GetFormatNames, GetPrintCounter, GetPrinterInfo, GetProperty, GetPropertyAttribute, GetPropertyNames, GetVariable, GetVariableNames, Lock, PrintFormat, Reboot, RenameFormat, ResetError, SendFormat, SetFormat, SetProperty, SetPropertyNames, SetVariable, Stop, TriggerPrint, Unlock, and Version.

Name	Value	Type	Base Type	Status	Server Timestamp	Source Timestamp	AccessLevel
GetConnectionStatus		Method			12.10.2022 08:16:26		None
GetErrorDetails		Method			12.10.2022 08:16:26		None
GetErrorStatus		Method			12.10.2022 08:16:26		None
GetFontNames		Method			12.10.2022 08:16:26		None
GetFormat		Method			12.10.2022 08:16:26		None
GetFormatNames		Method			12.10.2022 08:16:26		None
GetPrintCounter		Method			12.10.2022 08:16:26		None
GetPrinterInfo		Method			12.10.2022 08:16:26		None
GetProperty		Method			12.10.2022 08:16:26		None
GetPropertyAttribute		Method			12.10.2022 08:16:26		None
GetPropertyNames		Method			12.10.2022 08:16:26		None
GetVariable		Method			12.10.2022 08:16:26		None
GetVariableNames		Method			12.10.2022 08:16:26		None
Lock		Method			12.10.2022 08:16:26		None
PrintFormat		Method			12.10.2022 08:16:26		None
Reboot		Method			12.10.2022 08:16:26		None
RenameFormat		Method			12.10.2022 08:16:26		None
ResetError		Method			12.10.2022 08:16:26		None
SendFormat		Method			12.10.2022 08:16:26		None
SetFormat		Method			12.10.2022 08:16:26		None
SetProperty		Method			12.10.2022 08:16:26		None
SetPropertyNames		Method			12.10.2022 08:16:26		None
SetVariable		Method			12.10.2022 08:16:26		None
Stop		Method			12.10.2022 08:16:26		None
TriggerPrint		Method			12.10.2022 08:16:26		None
Unlock		Method			12.10.2022 08:16:26		None
Version	1.12.15801-R CA...	String		Good - The ope...	12.10.2022 08:16:26	12.10.2022 07:04:48	CurrentRead

13. Properties of the web box

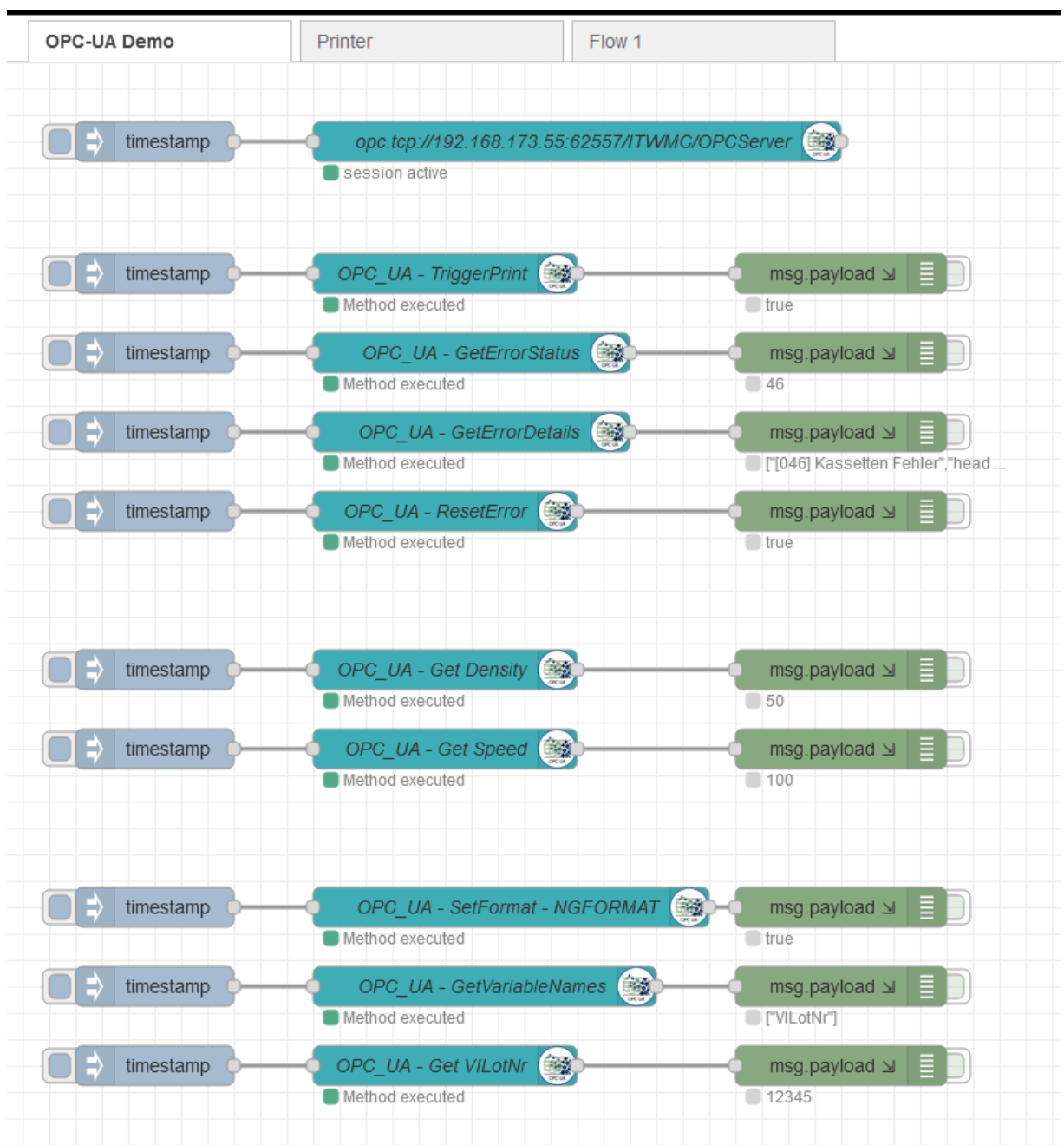
node name	node class	ID	description	JSON serialized return values
GetConnectionStatus	method	ns=2;i=14	Printer connection status	boolean
GetErrorDetails	method	ns=2;i=32	Reading the error details	List<string>
GetErrorStatus	method	ns=2;i=29	Reading the error status	string
GetFontNames	method	ns=2;i=41	Installed fonts	List<string>
GetFormat	method	ns=2;i=44	Read native printer format	string
GetFormatNames	method	ns=2;i=38	List of existing print images	List<string>
GetPrintCounter	method	ns=2;i=35	number of prints	long
GetPrinterInfo	method	ns=2;i=4	General printer information	List<KeyValuePair<string, string>>
GetProperty	method	ns=2;i=82	Read printer property	string
GetPropertyAttribute	method	ns=2;i=78	Read printer property attributes	string
GetPropertyNames	method	ns=2;i=75	List of printer properties	List<string>
GetVariable	method	ns=2;i=67	Read variable content	string
GetVariableNames	method	ns=2;i=64	Variable names of the current print format	List<string>
lock	method	ns=2;i=7	UI Lock	boolean

print format	method	ns=2;i=60	Load print image	string
reboot	method	ns=2;i=23	Restart printer	boolean
RenameFormat	method	ns=2;i=52	Rename print image	boolean
ResetError	method	ns=2;i=20	reset errors	boolean
SendFormat	method	ns=2;i=56	Send native binary print format to printer (Base64)	string
SetFormat	method	ns=2;i=48	Set print format as active	boolean
SetProperty	method	ns=2;i=86	write property	boolean
SetVariable	method	ns=2;i=71	Write variable content	boolean
stop	method	ns=2;i=26	to stop the printer	boolean
TriggerPrint	method	ns=2;i=17	trigger pressure	boolean
unlock	method	ns=2;i=11	Unlock UI	boolean
version	Property	ns=2;i=2	version	boolean

14. WebBox OPC demo with Node-Red

A NODE-Red instance runs on the WebBox, with which actions can be carried out on the printer via OPC as a demo

<http://192.168.173.55:1880>



Node 'OpcUa-Method' bearbeiten

Löschen

⚙️ Eigenschaften

Endpoint

Objectid

Methodid

Arg0 Name Type Typeld

Arg1 Name Type Typeld

Arg2 Name Type Typeld

Out0 Name Type Typeld

Name