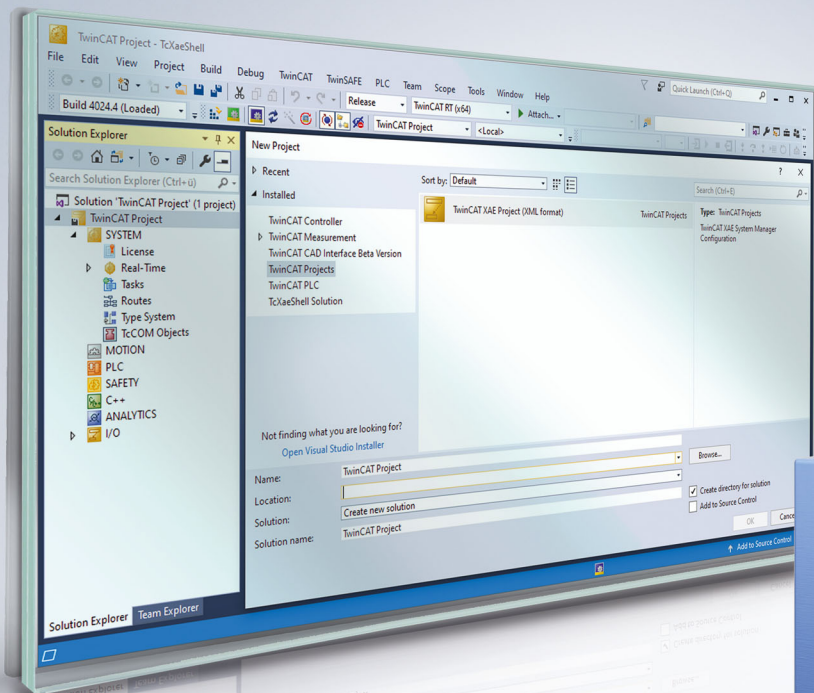


# BECKHOFF New Automation Technology

Handbuch | DE

## TF3520

TwinCAT 3 | Analytics Storage Provider





# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort.....</b>	<b>5</b>
1.1	Hinweise zur Dokumentation .....	5
1.2	Zu Ihrer Sicherheit.....	6
1.3	Hinweise zur Informationssicherheit .....	7
<b>2</b>	<b>Übersicht.....</b>	<b>8</b>
<b>3</b>	<b>Installation .....</b>	<b>9</b>
3.1	Systemvoraussetzungen .....	9
3.2	Installation .....	9
3.3	Lizenzierung .....	12
3.4	Installation TwinCAT/BSD .....	14
<b>4</b>	<b>Analytics Workflow – Erste Schritte.....</b>	<b>16</b>
4.1	Aufzeichnung von Maschinendaten .....	16
4.2	Kommunikation .....	19
4.3	Historisierung von Daten .....	20
4.4	Importieren von Analytics Files .....	27
4.5	Analyse der Daten.....	28
4.6	24-stündige Anwendung von Analytics .....	33
<b>5</b>	<b>Technische Einführung .....</b>	<b>43</b>
<b>6</b>	<b>Konfiguration.....</b>	<b>45</b>
6.1	Service .....	45
6.2	Datenbanken/Storage .....	47
6.2.1	Analytics Binary File.....	47
6.2.2	Microsoft SQL .....	49
6.2.3	Microsoft Azure Blob .....	50
6.3	Recorder .....	52
6.4	Arbeit mit historischen Daten .....	56
6.5	Konsolen- Konfigurator/Client .....	60
6.5.1	Konfigurator.....	61
6.5.2	Client.....	62
6.5.3	Batch Files zur Steuerung.....	65
<b>7</b>	<b>SPS API .....</b>	<b>69</b>
7.1	Funktionsbausteine .....	69
7.1.1	Topic-Architektur .....	69
7.1.2	FB_ALY_StorageProvider.....	82
7.2	Datentypen.....	88
7.2.1	ST_ConnectionSettings .....	88
7.2.2	ST_ALY_SP_Config.....	88
7.2.3	E_CancelType.....	89
7.2.4	E_RawDataFormat.....	89
7.2.5	E_RecordMode .....	90
7.2.6	E_ReloadType .....	90
7.2.7	E_RingBufferMode.....	90
7.2.8	E_SetGetHistoricalDataState.....	91

---

7.2.9	E_SymbolMode.....	91
<b>8</b>	<b>Beispiele .....</b>	<b>92</b>
8.1	SPS Client.....	92
<b>9</b>	<b>Anhang.....</b>	<b>97</b>
9.1	FAQ – Häufig gestellte Fragen und Antworten .....	97
9.2	Third-party components .....	98

# 1 Vorwort

## 1.1 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, stets die aktuell gültige Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

### Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiterentwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

### Marken

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

### Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.



EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizenziert durch die Beckhoff Automation GmbH, Deutschland

### Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

## 1.2 Zu Ihrer Sicherheit

### Sicherheitsbestimmungen

Lesen Sie die folgenden Erklärungen zu Ihrer Sicherheit.  
Beachten und befolgen Sie stets produktspezifische Sicherheitshinweise, die Sie gegebenenfalls an den entsprechenden Stellen in diesem Dokument vorfinden.

### Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

### Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

### Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden. Um Personen- und Sachschäden zu vermeiden, lesen und befolgen Sie die Sicherheits- und Warnhinweise.

### Warnungen vor Personenschäden

#### **GEFAHR**

Es besteht eine Gefährdung mit hohem Risikograd, die den Tod oder eine schwere Verletzung zur Folge hat.

#### **WARNUNG**

Es besteht eine Gefährdung mit mittlerem Risikograd, die den Tod oder eine schwere Verletzung zur Folge haben kann.

#### **VORSICHT**

Es besteht eine Gefährdung mit geringem Risikograd, die eine mittelschwere oder leichte Verletzung zur Folge haben kann.

### Warnung vor Umwelt- oder Sachschäden

#### **HINWEIS**

Es besteht eine mögliche Schädigung für Umwelt, Geräte oder Daten.

### Information zum Umgang mit dem Produkt



Diese Information beinhaltet z. B.:  
Handlungsempfehlungen, Hilfestellungen oder weiterführende Informationen zum Produkt.

## 1.3 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

## 2 Übersicht

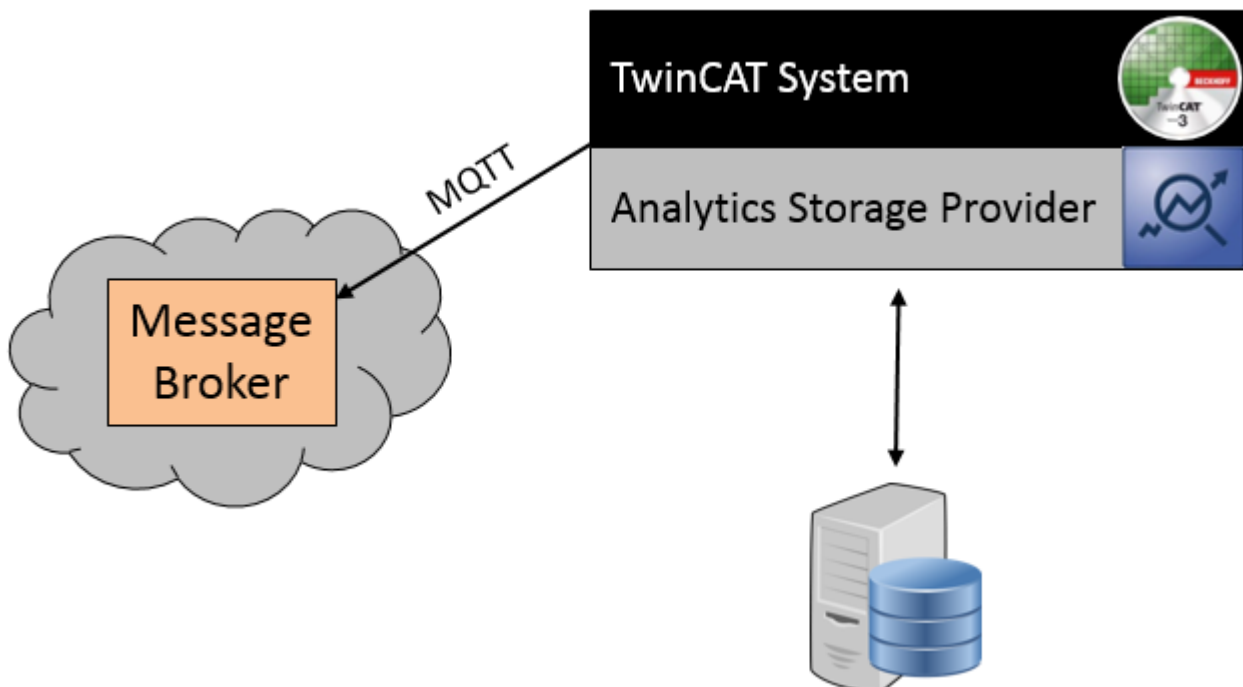
Beckhoff bietet mit dem TwinCAT Analytics Storage Provider eine Möglichkeit, hochaufgelöste Daten in einem Binärformat kontinuierlich abzulegen. Entscheidend ist, dass sich der Anwender keine Gedanken zur Datenablage machen muss. Darum kümmert sich der Storage Provider automatisch. Die Konfiguration erfolgt mit wenigen Klicks im Engineering. Aufwendige SQL-Befehle sind nicht notwendig. Es können klassische Datenbanken, aber auch binäre Blob-Stores verwendet werden.

### Komponenten

- TwinCAT Analytics Storage Provider Service: Ein Windows Service, der die Kommunikation verwaltet.
- TwinCAT Analytics Storage Provider PLC Library: Eine TwinCAT 3 SPS-Bibliothek mit Funktionen zur Steuerung des Storage Providers aus einer SPS-Anwendung.

### Funktionsprinzip

Der Analytics Storage Provider empfängt und sendet Daten via MQTT Kommunikationsprotokoll. Dazu ist er mit einem nativen MQTT-Message-Broker im Netzwerk verbunden und auf der anderen Seite mit der entsprechenden Datenbank.



### Unterstützte Datenbanken/Storage

- TwinCAT Analytics Binary File
- Microsoft SQL
- Microsoft Azure Blob



### 3 Installation

#### 3.1 Systemvoraussetzungen

Die Anforderungen des Service und der PLC Library des Analytics Storage Providers finden Sie in den folgenden Tabellen. Beide können auch auf einem System installiert werden.

Technische Daten Service	TF3520 TC 3 Analytics Storage Provider
Zielsystem	Windows 7, Windows 8, Windows 10
.NET Framework	.Net 4.5.1 oder höher
Min. TwinCAT-Version	3.1.4022.25
Min. TwinCAT Level	TC1000 TC3   ADS

Technische Daten Library	TF3520 TC 3 Analytics Storage Provider
Zielsystem	Windows 7, Windows 8, Windows 10
Min. TwinCAT-Version	3.1.4022.29
Min. TwinCAT Level	TC1200 TC3   PLC

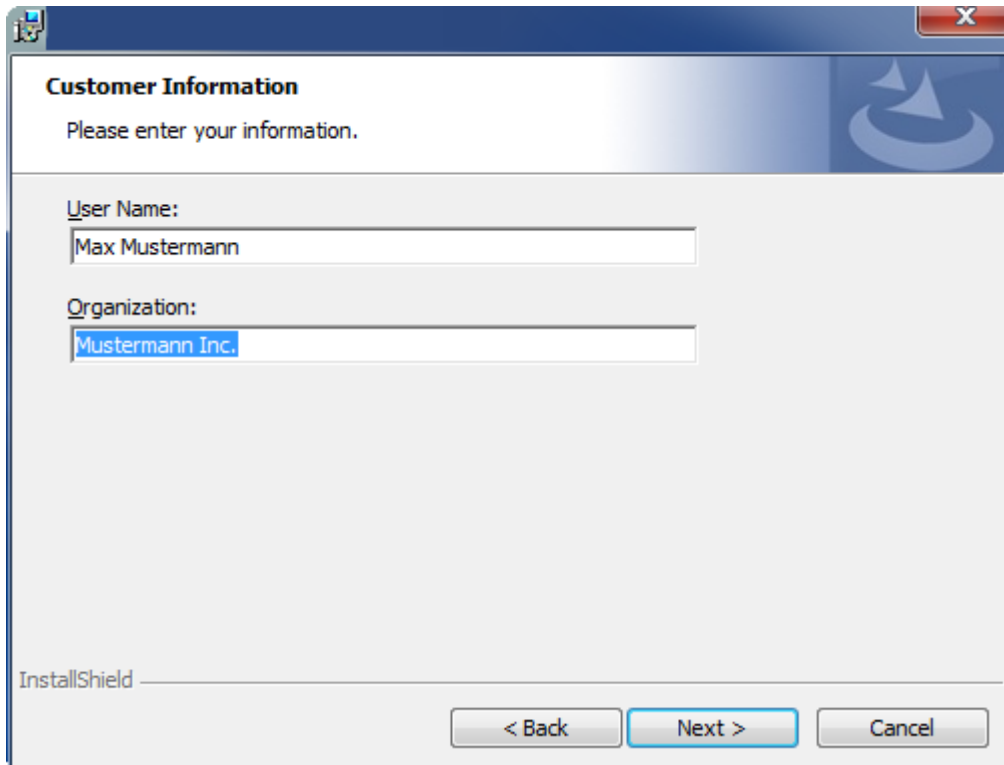
#### 3.2 Installation

Nachfolgend wird beschrieben, wie die TwinCAT 3 Function für Windows-basierte Betriebssysteme installiert wird.

- ✓ Die Setup-Datei der TwinCAT 3 Function wurde von der Beckhoff-Homepage heruntergeladen.
- 1. Führen Sie die Setup-Datei als Administrator aus. Wählen Sie dazu im Kontextmenü der Datei den Befehl **Als Administrator ausführen**.
  - ⇒ Der Installationsdialog öffnet sich.
- 2. Akzeptieren Sie die Endbenutzerbedingungen und klicken Sie auf **Next**.



3. Geben Sie Ihre Benutzerdaten ein.



**Customer Information**  
Please enter your information.

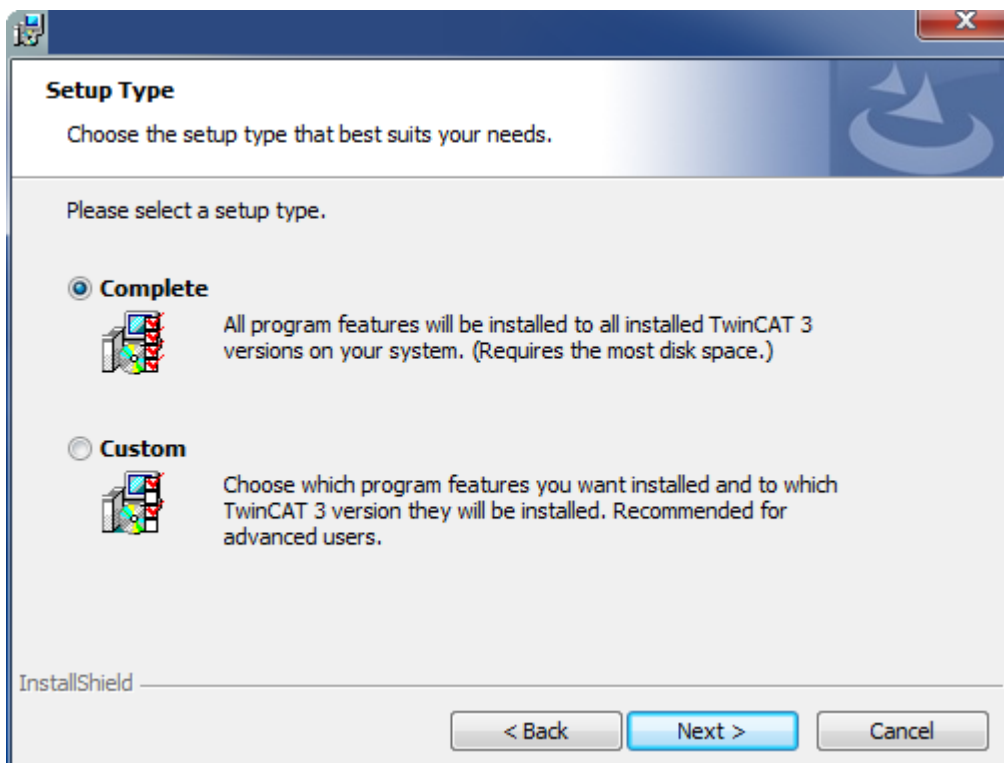
User Name:  
Max Mustermann

Organization:  
Mustermann Inc.

InstallShield


< Back   Next >   Cancel


4. Wenn Sie die TwinCAT 3 Function vollständig installieren möchten, wählen Sie **Complete** als Installationstyp. Wenn Sie die Komponenten der TwinCAT 3 Function separat installieren möchten, wählen Sie **Custom**.



**Setup Type**  
Choose the setup type that best suits your needs.

Please select a setup type.

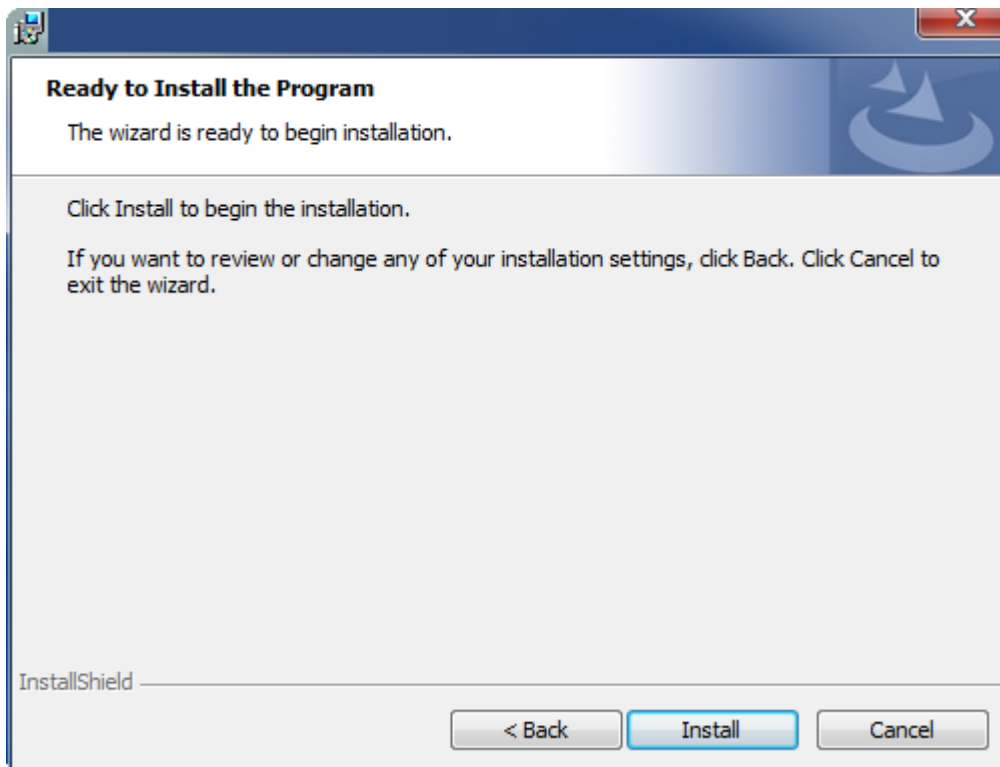
**Complete**  
 All program features will be installed to all installed TwinCAT 3 versions on your system. (Requires the most disk space.)

**Custom**  
 Choose which program features you want installed and to which TwinCAT 3 version they will be installed. Recommended for advanced users.

InstallShield

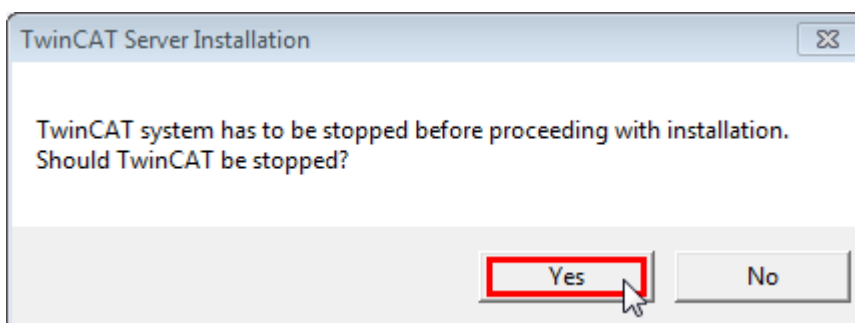
< Back   Next >   Cancel

5. Wählen Sie **Next** und anschließend **Install**, um die Installation zu beginnen.

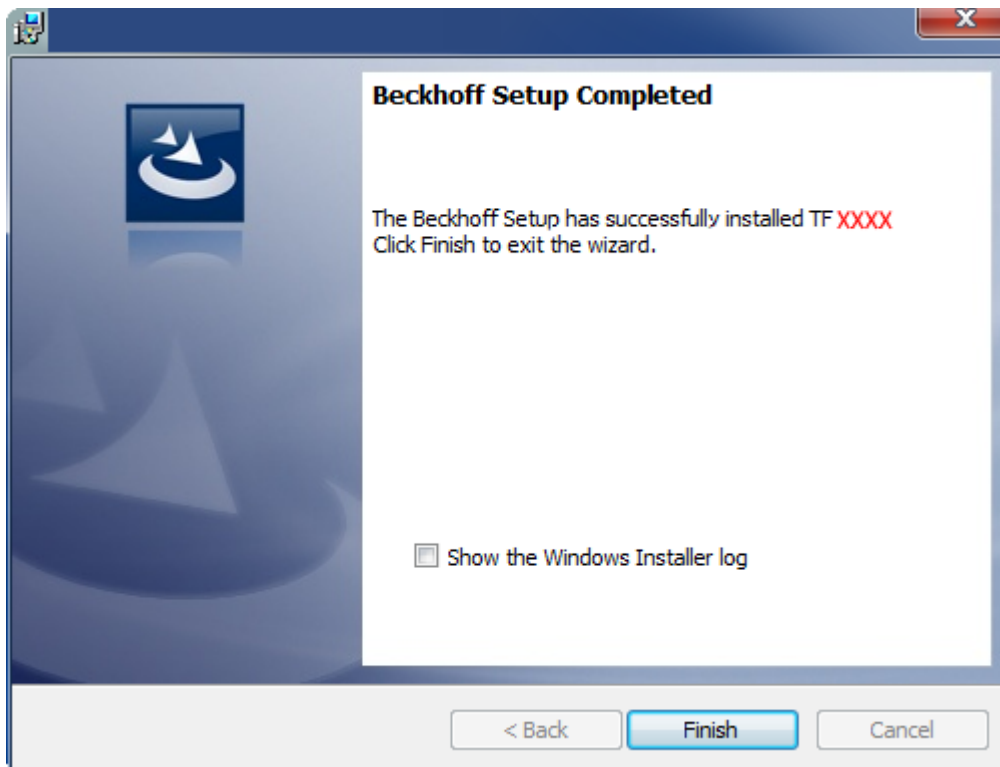


⇒ Ein Dialog weist Sie darauf hin, dass das TwinCAT-System für die weitere Installation gestoppt werden muss.

6. Bestätigen Sie den Dialog mit **Yes**.



7. Wählen Sie **Finish**, um das Setup zu beenden.



⇒ Die TwinCAT 3 Function wurde erfolgreich installiert und kann lizenziert werden (siehe [Lizenzierung \[► 12\]](#)).

### 3.3 Lizenzierung

Die TwinCAT 3 Function ist als Vollversion oder als 7-Tage-Testversion freischaltbar. Beide Lizenztypen sind über die TwinCAT-3-Entwicklungsumgebung (XAE) aktivierbar.

#### Lizenzierung der Vollversion einer TwinCAT 3 Function

Die Beschreibung der Lizenzierung einer Vollversion finden Sie im Beckhoff Information System in der Dokumentation „[TwinCAT 3 Lizenzierung](#)“.

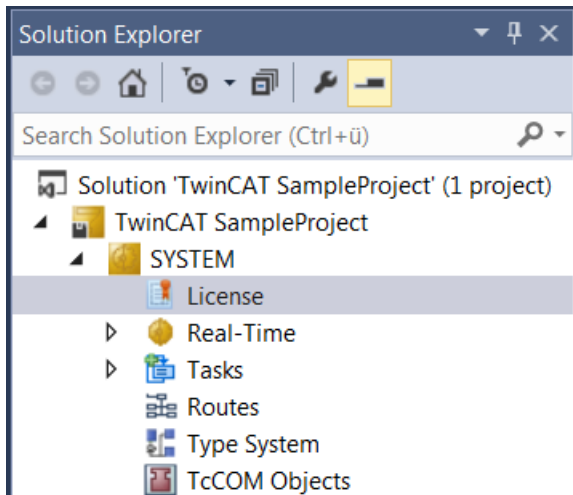
#### Lizenzierung der 7-Tage-Testversion einer TwinCAT 3 Function



Eine 7-Tage-Testversion kann nicht für einen [TwinCAT-3-Lizenz-Dongle](#) freigeschaltet werden.

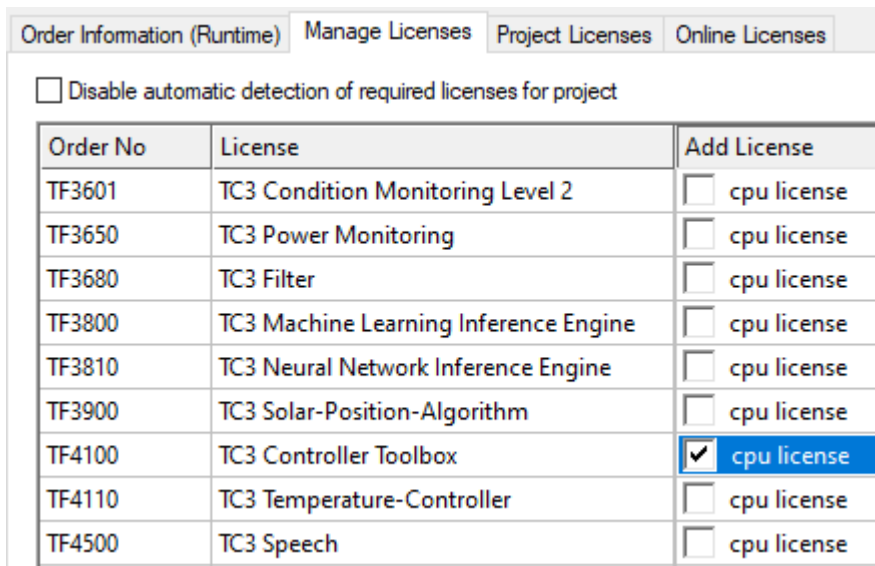
1. Starten Sie die TwinCAT-3-Entwicklungsumgebung (XAE).
2. Öffnen Sie ein bestehendes TwinCAT-3-Projekt oder legen Sie ein neues Projekt an.
3. Wenn Sie die Lizenz für ein Remote-Gerät aktivieren wollen, stellen Sie das gewünschte Zielsystem ein. Wählen Sie dazu in der Symbolleiste in der Drop-down-Liste **Choose Target System** das Zielsystem aus.
  - ⇒ Die Lizenzierungseinstellungen beziehen sich immer auf das eingestellte Zielsystem. Mit der Aktivierung des Projekts auf dem Zielsystem werden automatisch auch die zugehörigen TwinCAT-3-Lizenzen auf dieses System kopiert.

4. Klicken Sie im **Solution Explorer** im Teilbaum **SYSTEM** doppelt auf **License**.



⇒ Der TwinCAT-3-Lizenzmanager öffnet sich.

5. Öffnen Sie die Registerkarte **Manage Licenses**. Aktivieren Sie in der Spalte **Add License** das Auswahlkästchen für die Lizenz, die Sie Ihrem Projekt hinzufügen möchten (z. B. „TF4100 TC3 Controller Toolbox“).



6. Öffnen Sie die Registerkarte **Order Information (Runtime)**.

⇒ In der tabellarischen Übersicht der Lizenzen wird die zuvor ausgewählte Lizenz mit dem Status „missing“ angezeigt.

7. Klicken Sie auf **7 Days Trial License...**, um die 7-Tage-Testlizenz zu aktivieren.

The screenshot shows a software interface for license management. At the top, there are tabs: 'Order Information (Runtime)', 'Manage Licenses', 'Project Licenses', and 'Online Licenses'. Below these are three main sections:

- License Device:** Includes a dropdown menu set to 'Target (Hardware Id)', an 'Add...' button, a 'System Id:' field containing '2DB25408-B4CD-81DF-5488-6A3D9B49EF19', and a 'Platform:' dropdown menu set to 'other (91)'.
- License Request:** Includes a 'Provider:' dropdown menu set to 'Beckhoff Automation', a 'Generate File...' button, 'License Id:' and 'Customer Id:' input fields, and a 'Comment:' text area.
- License Activation:** Contains two buttons: '7 Days Trial License...' (highlighted with a red rectangle) and 'License Response File...'.

⇒ Es öffnet sich ein Dialog, der Sie auffordert, den im Dialog angezeigten Sicherheitscode einzugeben.

The screenshot shows a dialog box titled 'Enter Security Code'. It contains the text 'Please type the following 5 characters:'. Below this text is a box displaying the code 'Kg8T4'. To the right of the code box is an 'OK' button, which is highlighted with a red rectangle. Below the code box is another empty box with a red border, and to its right is a 'Cancel' button.

8. Geben Sie den Code genauso ein, wie er angezeigt wird, und bestätigen Sie ihn.

9. Bestätigen Sie den nachfolgenden Dialog, der Sie auf die erfolgreiche Aktivierung hinweist.

⇒ In der tabellarischen Übersicht der Lizenzen gibt der Lizenzstatus nun das Ablaufdatum der Lizenz an.

10. Starten Sie das TwinCAT-System neu.

⇒ Die 7-Tage-Testversion ist freigeschaltet.

### 3.4 Installation TwinCAT/BSD

Der TwinCAT 3 Analytics Storage Provider Server steht als Package für TwinCAT/BSD im Package-Repository bereit. Unter dem Package Namen „TF3520-Analytics-Storage-Provider“ kann er über folgenden Befehl installiert werden:

```
doas pkg install TF3520-Analytics-Storage-Provider
```

Weitere Informationen über den [Package Server](#) finden Sie im Bereich Embedded-PC im TwinCAT/BSD-Handbuch.

Nach einem Neustart des Systems oder Restart von TwinCAT, wird der TwinCAT 3 Analytics Storage Provider Server ebenfalls gestartet und ist über MQTT von einem Client konfigurierbar.

#### **i** MQTT-Port-Freigabe

Für die Verwendung des Analytics Storage Providers und des [Konsolen- Konfigurator/Client](#) [▶ 60] unter TwinCAT/BSD muss der entsprechende MQTT-Port für die Kommunikation freigegeben werden. Weitere Infos siehe: [Portfreigabe unter TwinCAT/BSD](#)

Nach der Installation befindet sich die Client.dll für die Konsole unter dem Pfad `/usr/local/etc/TwinCAT/Functions/TF3520-Analytics-Storage-Provider/Client`.

Der Analytics Storage Provider Service kann bei aktivierter Lizenz durch folgenden Befehl gestartet werden:

```
doas service TcAnalyticsStorageProvider start
```

## 4 Analytics Workflow – Erste Schritte

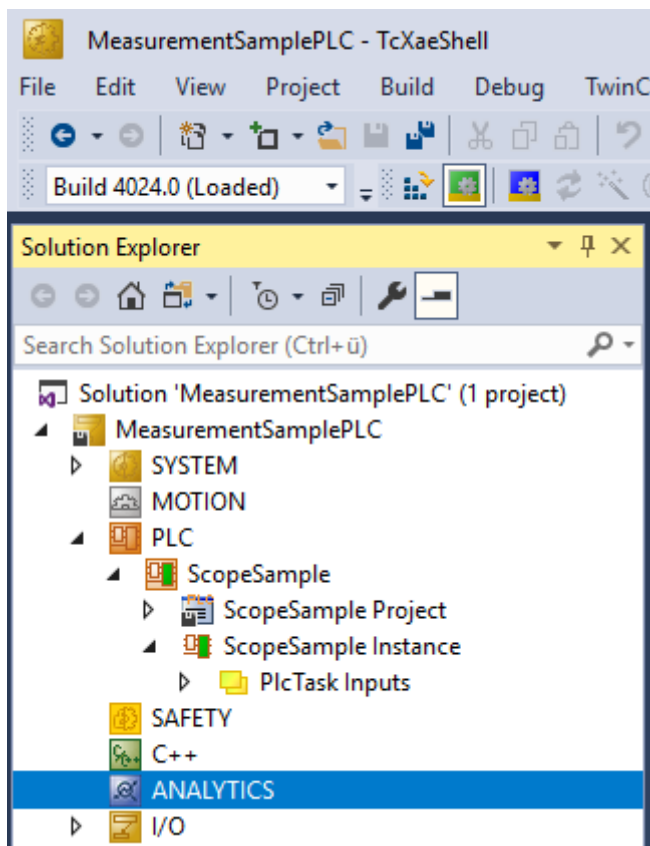
Diese Dokumentation präsentiert Schritt für Schritt den vollständigen TwinCAT Analytics Workflow. Von der Datenerfassung über die Kommunikation und Historisierung bis hin zur Auswertung und Analyse der Daten sowie zur Präsentation der Daten im webbasierten Dashboard.

### 4.1 Aufzeichnung von Maschinendaten

Maschinenseitig ist der Analytics Logger die Aufzeichnungseinrichtung von Prozessdaten des Maschinenabbaus, SPS, NC usw. Der Logger arbeitet im Echtzeitkontext von TwinCAT.

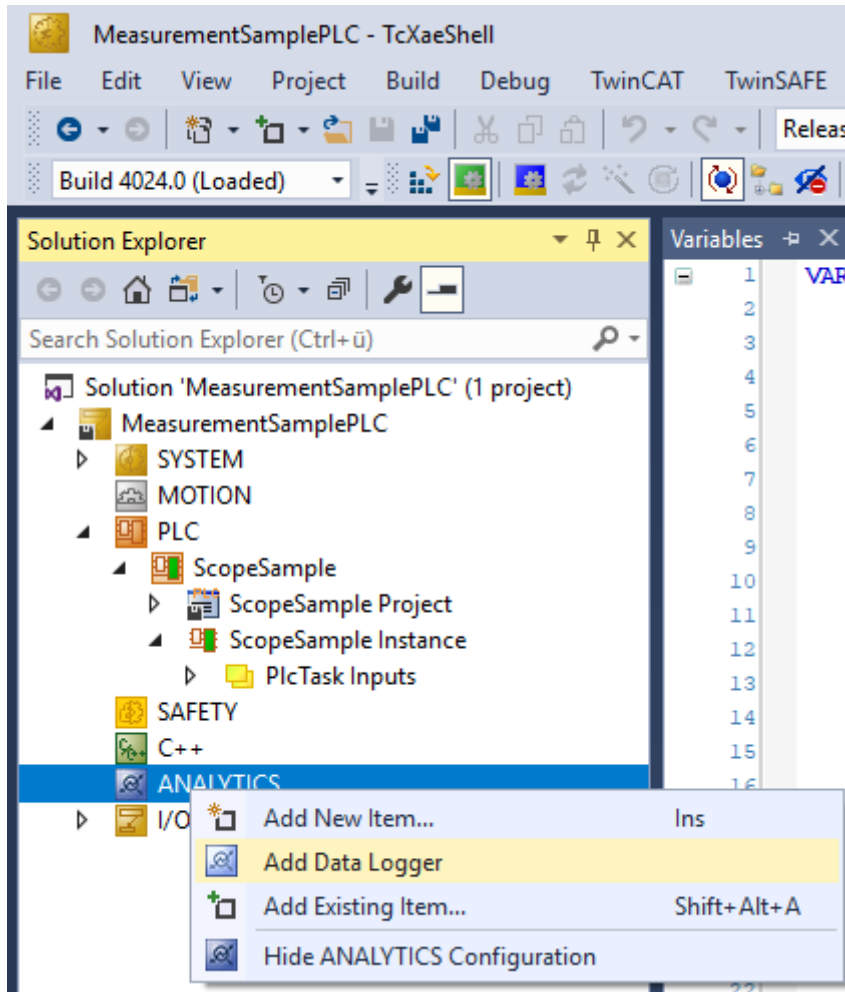
Der TwinCAT Analytics Logger wird mit TwinCAT XAE und XAR installiert. Der Logger kann als MQTT-Client fungieren, um die aufgezeichneten Daten an einen nativen MQTT-Message-Broker zu übermitteln, oder die Daten im selben Datenformat in einer lokalen Binärdatei zu speichern. Bei Verwendung als MQTT-Client kann der Logger kurze Unterbrechungen der Verbindung zum Message-Broker mit Hilfe einer Ringpuffer-Funktionalität überbrücken. Auch für den lokalen Binärdateispeicher kann ein Ringpuffer konfiguriert werden.

- Um den Analytics Logger zu konfigurieren, müssen Sie in Ihrem vorhandenen TwinCAT-Projekt zum Analytics-Baumknoten navigieren.

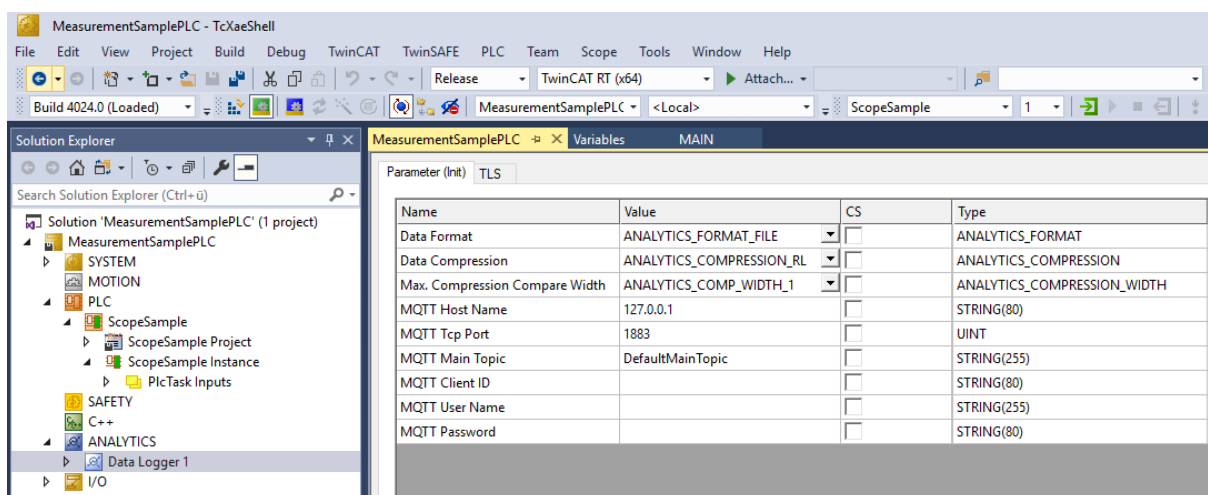




- Klicken Sie mit der rechten Maustaste auf diesen Knoten und klicken Sie auf **Add Data Logger**, um Ihrer Konfiguration eine neue Instanz hinzuzufügen.



- Für die Konfiguration der Grundeinstellungen doppelklicken Sie auf das neue Tree Item.



Sie können Ihre spezifischen Analytics Logger-Einstellungen vornehmen.

-Datenformat: Binärdatei oder MQTT-Stream.

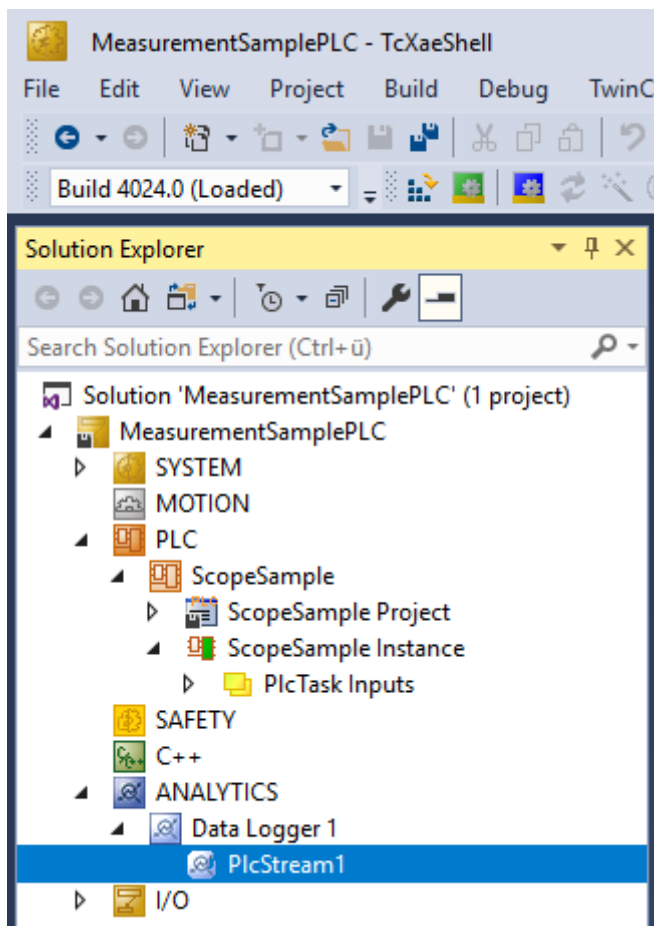
-FILE-Format: Der Analytics Logger speichert die Daten in lokalen Binärdateien und alle anderen Einstellungen sind nicht mehr notwendig. Die Dateien werden in *C:\TwinCAT\3.1\Boot\Analytics* gespeichert.

-BINARY-Format: Die Daten werden an den konfigurierten MQTT-Message-Broker gesendet. Sie können mehrere Logger in einem TwinCAT-Projekt vorsehen, um Daten an verschiedene MQTT-Message-Broker zu übermitteln.

- Datenkompression: ein (Standard) oder aus.
- Max. Kompression: Modus der Kompression.
- MQTT-Hostname
- MQTT-Tcp-Port
- MQTT-Maintopic für eigene Hierarchieebenen, damit die Identifikation einfach bleibt.
- MQTT-Client-ID, sollte im Netzwerk eindeutig sein.
- MQTT-Benutzername
- MQTT-Passwort für die Authentifizierung beim Message-Broker.

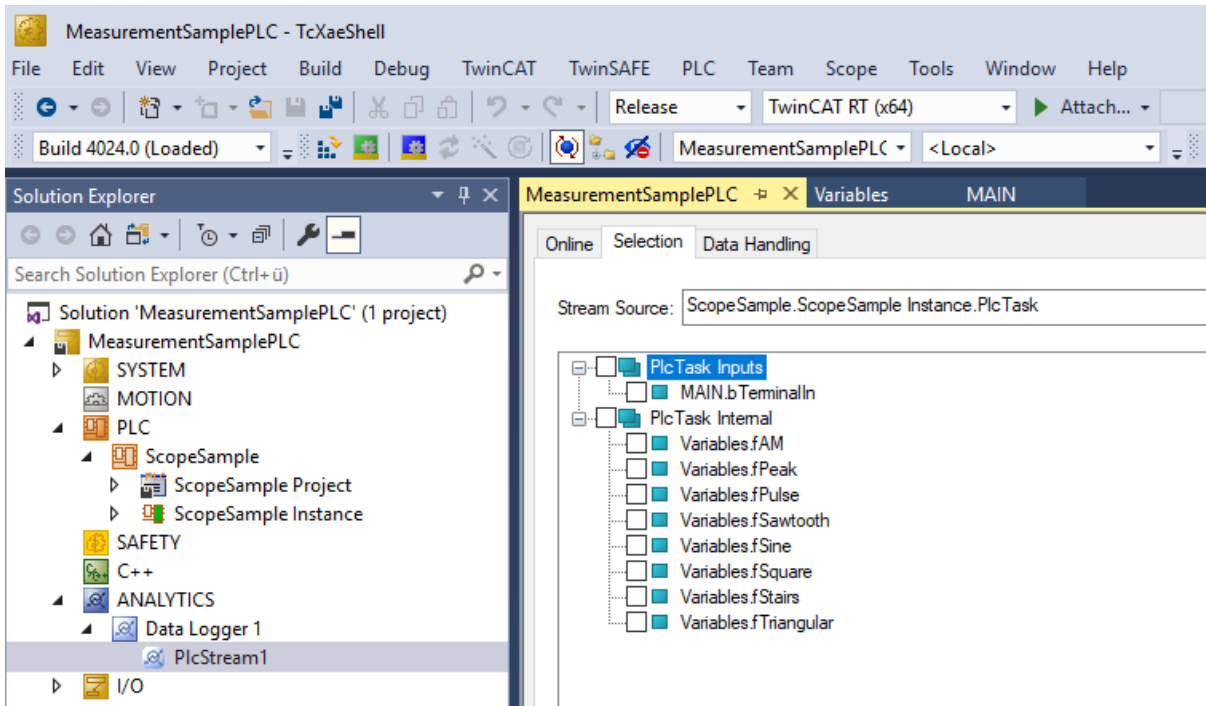
-Auf der Registerkarte **TLS** (Transport Layer Security) können die Sicherheitseinstellungen konfiguriert werden. TLS ist ein sicherer Kommunikationskanal zwischen Client und Server. Bei Verwendung von Zertifikaten ist der TCP-Port 8883 ausschließlich für MQTT über TLS reserviert! Der Analytics Logger unterstützt die Modi CA Certificates, CA Certificates & Client Certificate und Preshared Key (PSK).

- Wenn Variablen in Ihrer SPS-Anwendung in der Deklaration mit dem Attribut {attribute 'TcAnalytics'} gekennzeichnet sind, werden sie automatisch als Stream unter dem Daten-Logger-Baumknoten angezeigt.

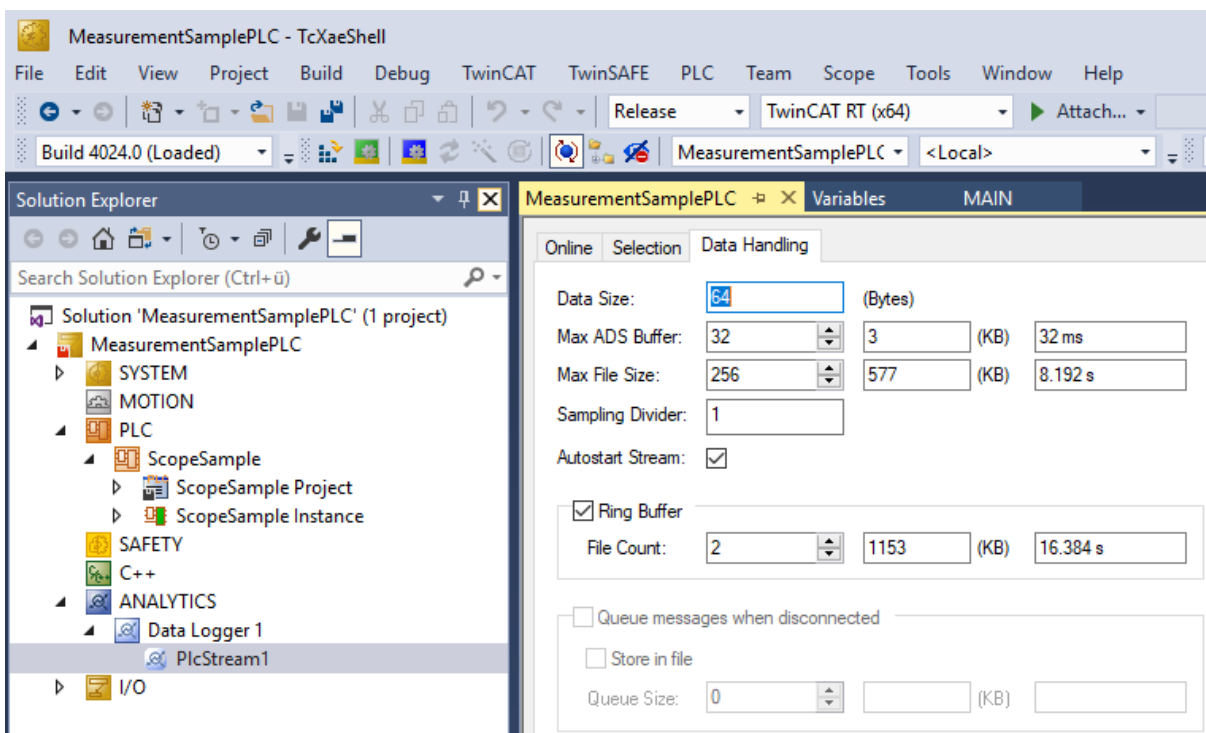


Ein zusätzlicher Geräte-Stream wird angezeigt, wenn Ihre Konfiguration ein EtherCAT-Prozessabbild liefert.

- In dem Stream gibt es einen Karteireiter **Selection**, wo die aufzuzeichnenden Variablen ausgewählt werden können.

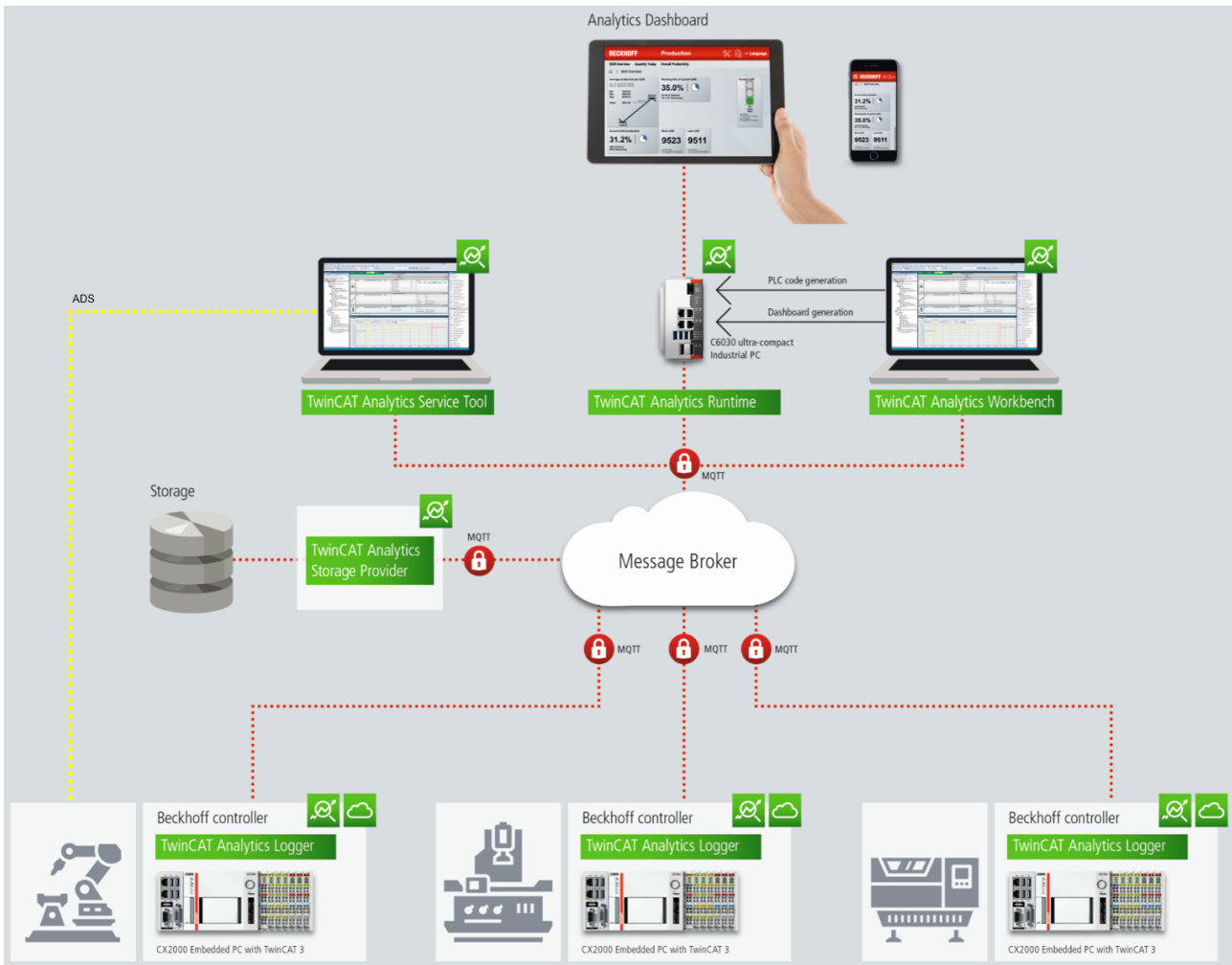


- Schließlich ist es im Karteireiter **Data Handling** möglich, die Paketgröße für die Frames zu ändern oder den Ringpuffer für Verbindungsunterbrechungen und Datei zu konfigurieren.



## 4.2 Kommunikation

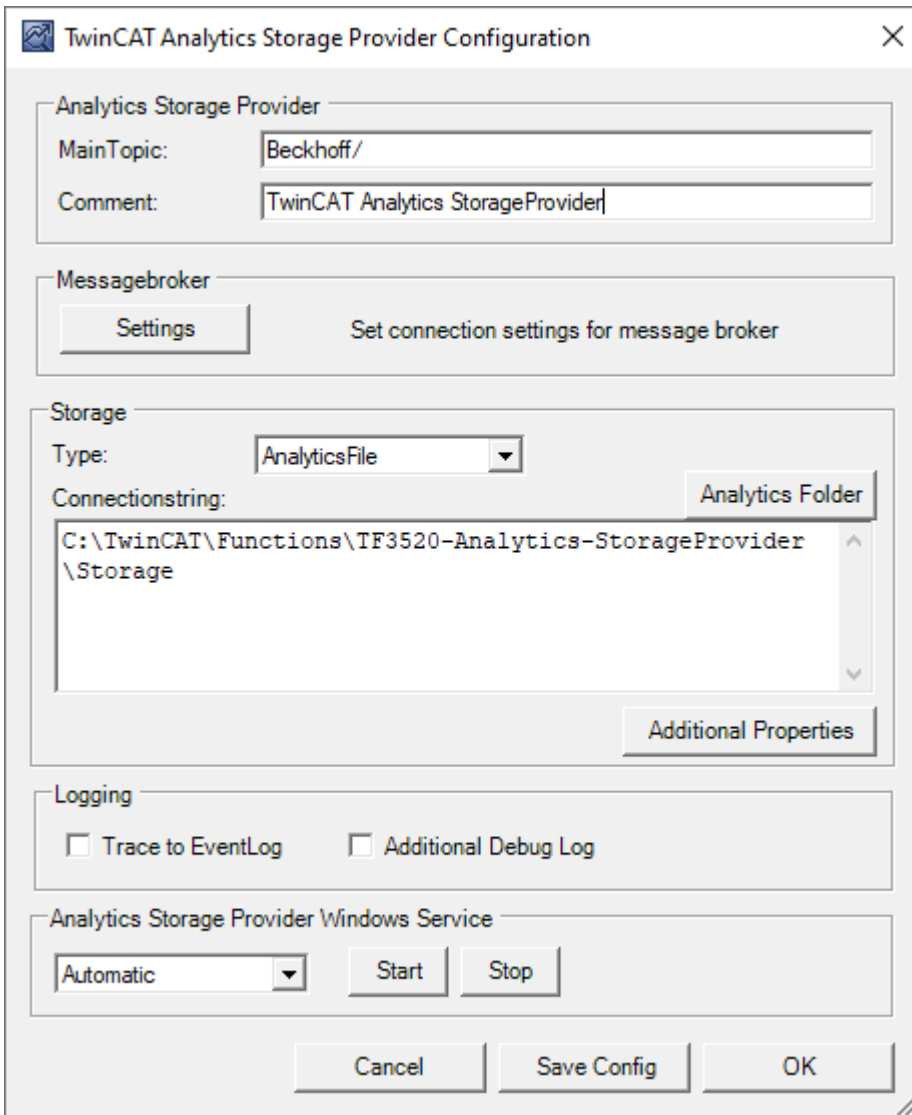
Derzeit kann der Analytics-Workflow vollständig über MQTT abgebildet werden. Die Engineering-Tools können auch über ADS auf die Daten der Maschinen zugreifen und Analysen durchführen.



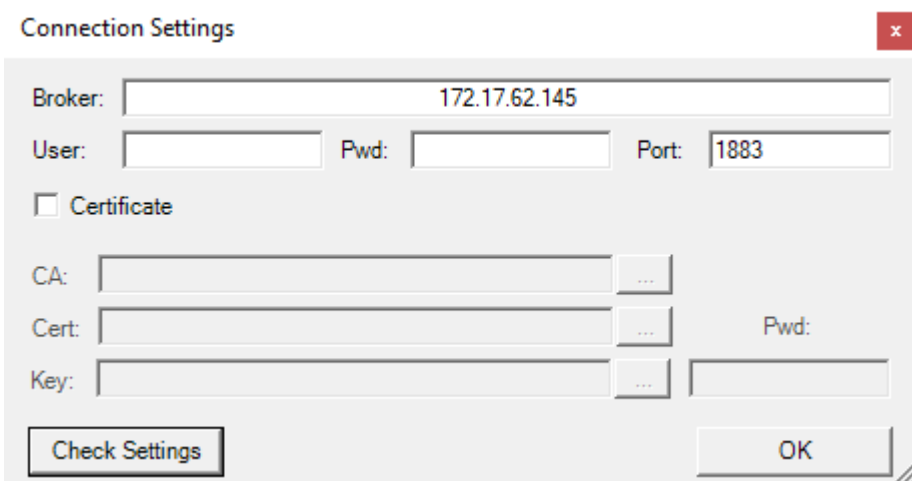
Wenn man für das IoT-Kommunikationsprotokoll MQTT wählt, muss irgendwo im Netzwerk ein nativer MQTT-Message-Broker eingerichtet werden (VM in einem Cloud-System ist ebenfalls möglich). Dieser Message-Broker sorgt für eine Entkopplung der verschiedenen Anwendungen im Analytics-Workflow.

### 4.3 Historisierung von Daten

Nach der Installation des TwinCAT Analytics Storage Provider kann der im Hintergrund laufende Dienst konfiguriert werden. Hierzu finden Sie die Anwendung TcAnalyticsStorageProvider\_Config im Ordner C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider\WinService.



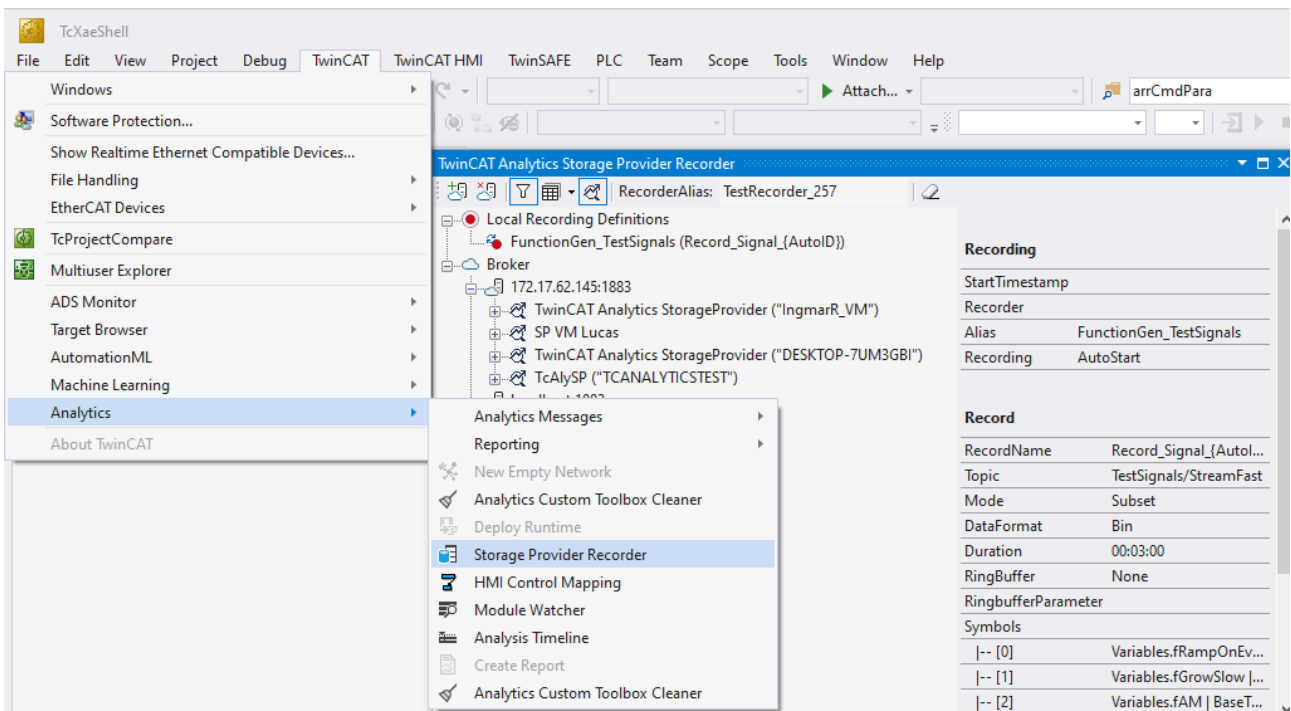
Der Hauptteil des Topics kann in der Konfiguration ebenso festgelegt werden wie der Kommentar, der zur Identifizierung dient, wenn mehr als ein Storage Provider beim Message-Broker registriert ist.



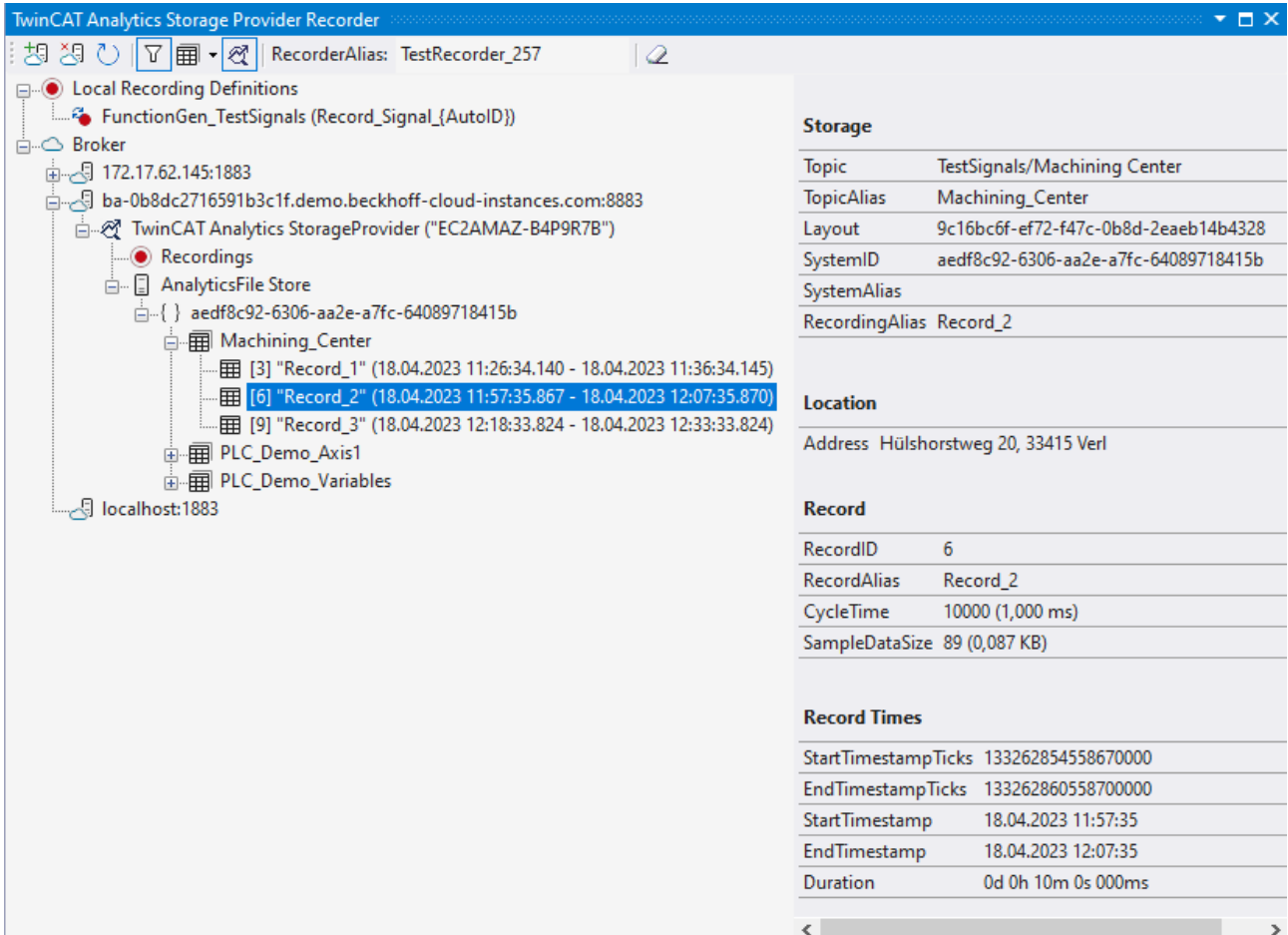
Sie können die Message-Broker-Einstellungen vornehmen und sich für einen Speichertyp entscheiden:

- Analytics File (Binärdatei)
- Microsoft SQL (Binär)
- Microsoft Azure Blob (Azure-Cloud erforderlich)

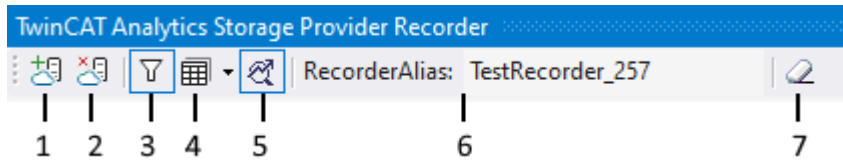
Zuletzt können Sie die Konfiguration speichern und den Dienst starten. Der nächste Schritt besteht in der Konfiguration der spezifischen Aufzeichnung. Hierzu sollten Sie in Ihrer Entwicklungsumgebung den **Storage Provider Recorder** auswählen.



Mit dem Storage Provider Recorder können Aufnahme Definitionen erstellt, gestartet und verwaltet werden. Zusätzlich besteht die Möglichkeit, die Datenspeicher einzelner Analytics Storage Provider zu verwalten. Alle wichtigen Eigenschaften der gefundenen Analytics Storage Providern und historisierten Daten werden übersichtlich dargestellt.



**Symbolleiste**

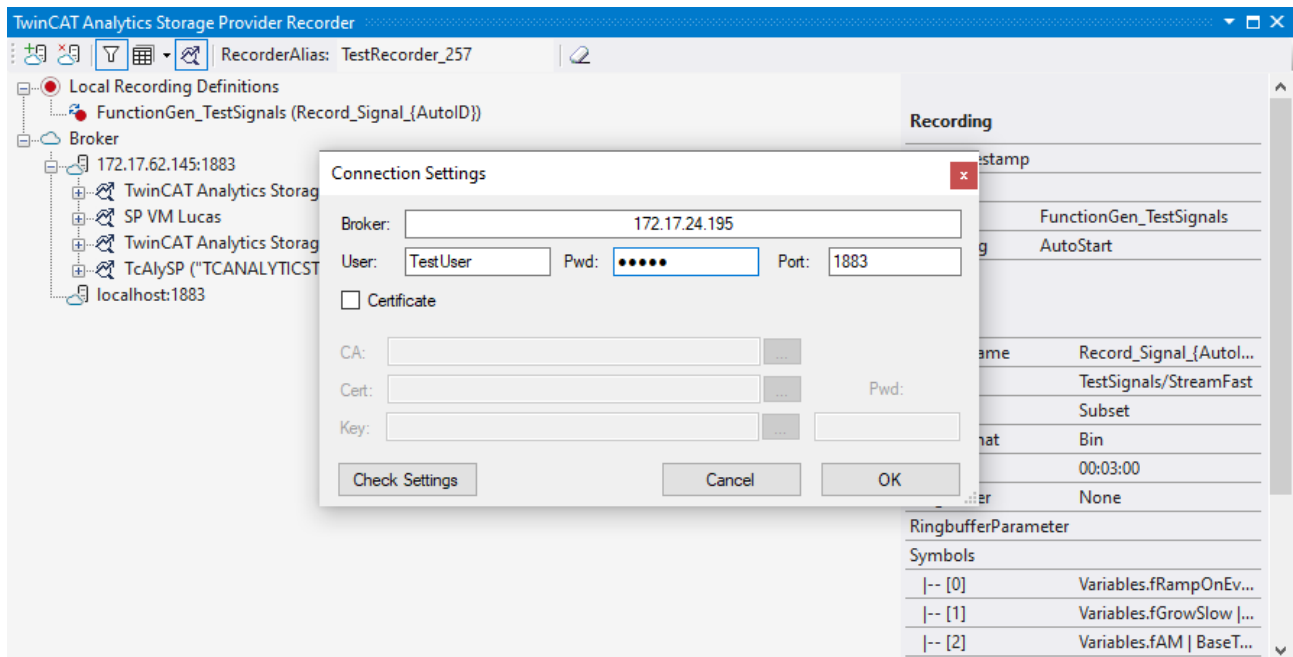


1	Neuen Broker hinzufügen
2	Ausgewählten Broker entfernen
3	Anzeige Filter (Alle / Meine aktiven Aufnahmen anzeigen)
4	Anzeige Art wählen (Alias / Topics)
5	Offline Analytics Storage Provider anzeigen / ausblenden
6	„RecorderAlias“ – Gruppierungsname von Aufnahmen
7	Meldungen aus Fehlerliste entfernen

**Recorder Fenster einrichten**

Vergeben Sie zunächst einen „RecorderAlias“. Dieser hilft die gestarteten Aufnahmen zu gruppieren und seine selbst gestarteten wieder zu finden. Mit dem Filter können auch fremd gestartete Aufnahmen angezeigt werden.

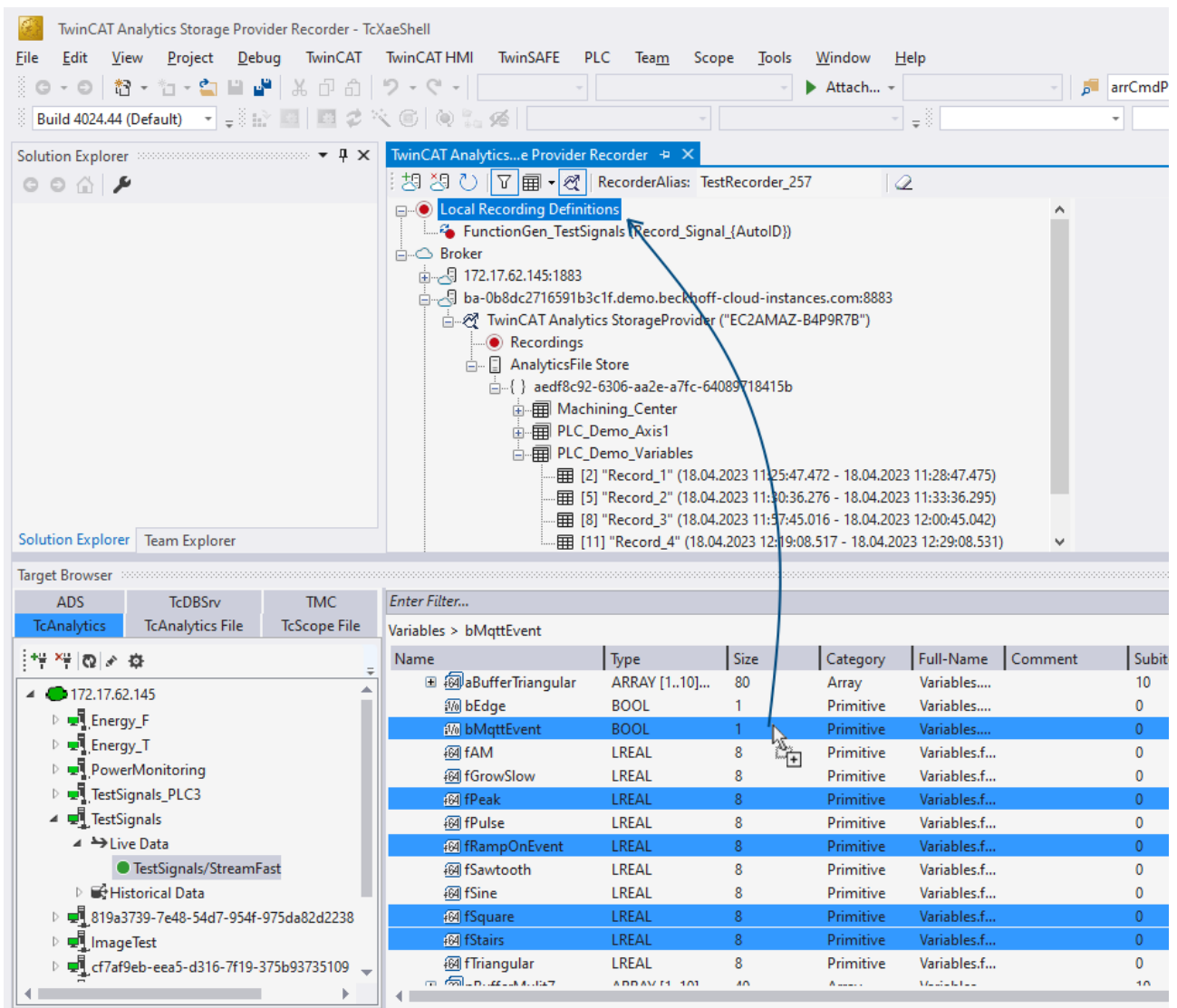
Danach können ein oder mehrere Broker eingerichtet werden. Dies geschieht über die schon bekannte Eingabemaske für MQTT-Verbindungseigenschaften.



Sobald eine Verbindung zum Broker hergestellt werden konnte, werden alle darauf verbundenen Analytics Storage Provider aufgelistet.

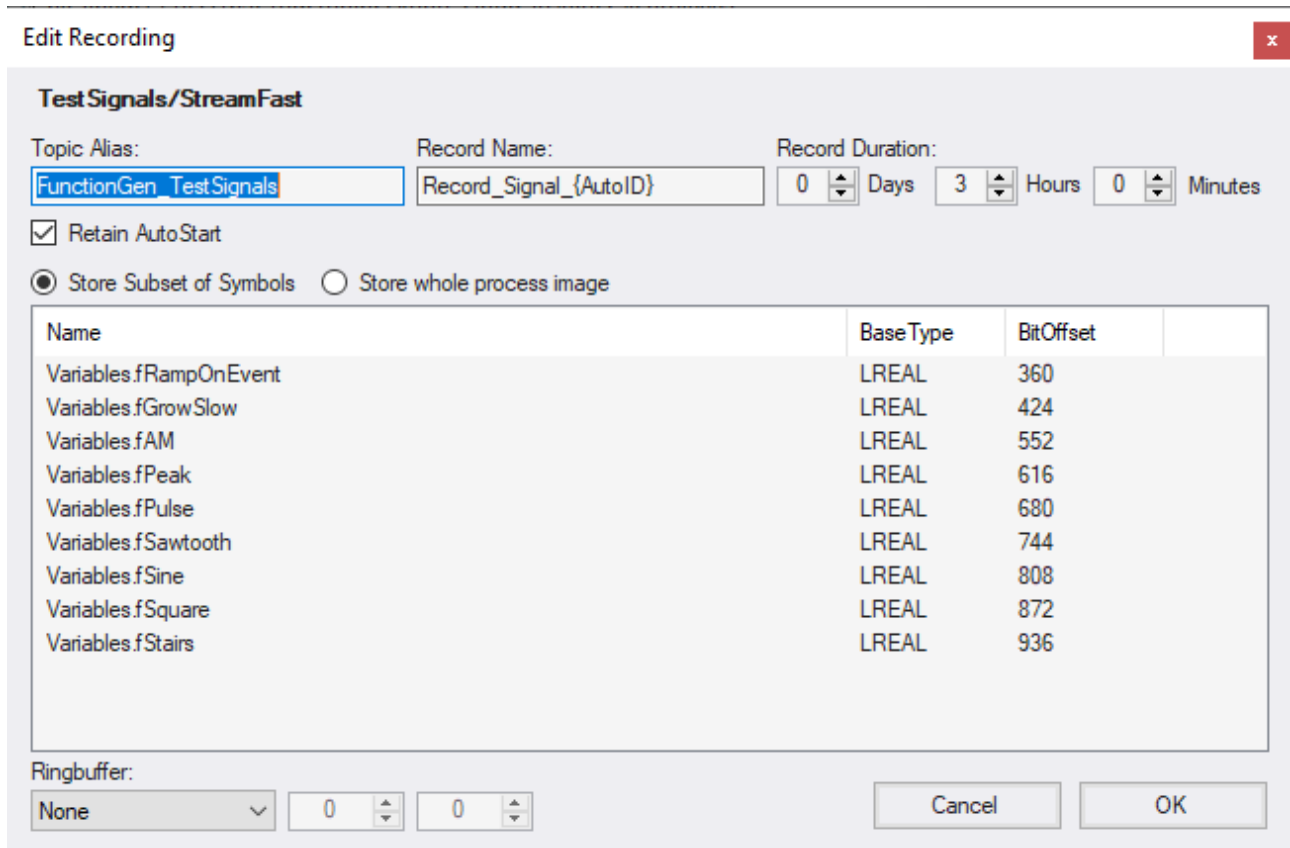
**Aufnahmedefinition erzeugen**

Zur Konfiguration der Aufzeichnung wählen Sie Ihr Ziel im Target Browser aus. Klicken Sie auf **Live Data** und wählen Sie eine oder mehrere Variablen durch Mehrfachauswahl aus und ziehen Sie sie per Drag-and-Drop auf den **Local Recording Definitions** Knoten im Recorder Fenster.



Im Recorder können Sie die ausgewählten Variablen oder das vollständige Quell-Prozessabbild der Variablen hinzufügen.



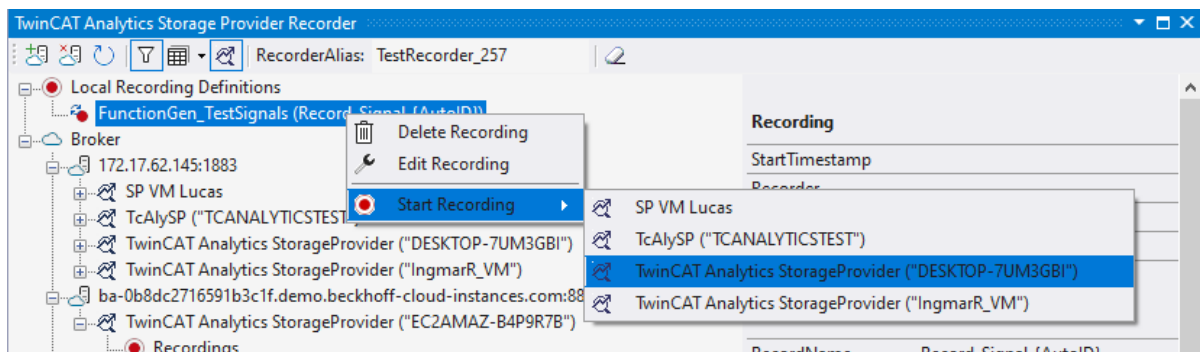


Sie können auch Aufzeichnungsamen und eine Dauer konfigurieren (anderenfalls läuft die Aufzeichnung endlos, bis sie manuell gestoppt wird). Ein Ringpuffer kann nach Storage oder Zeit eingestellt werden.

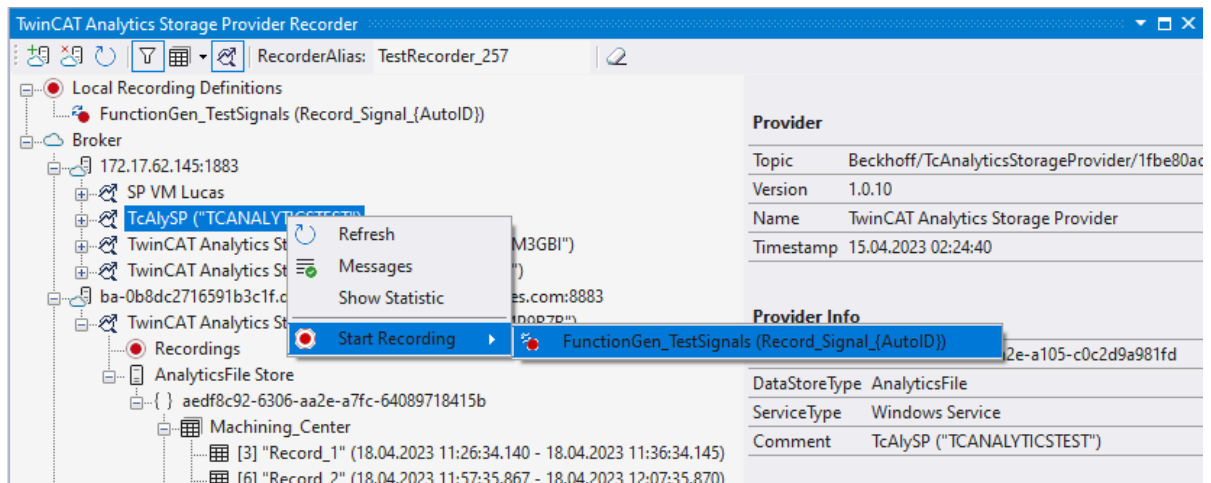
**Aufnahme starten**

Es gibt drei verschiedene Wege, um eine Aufnahme zu starten.

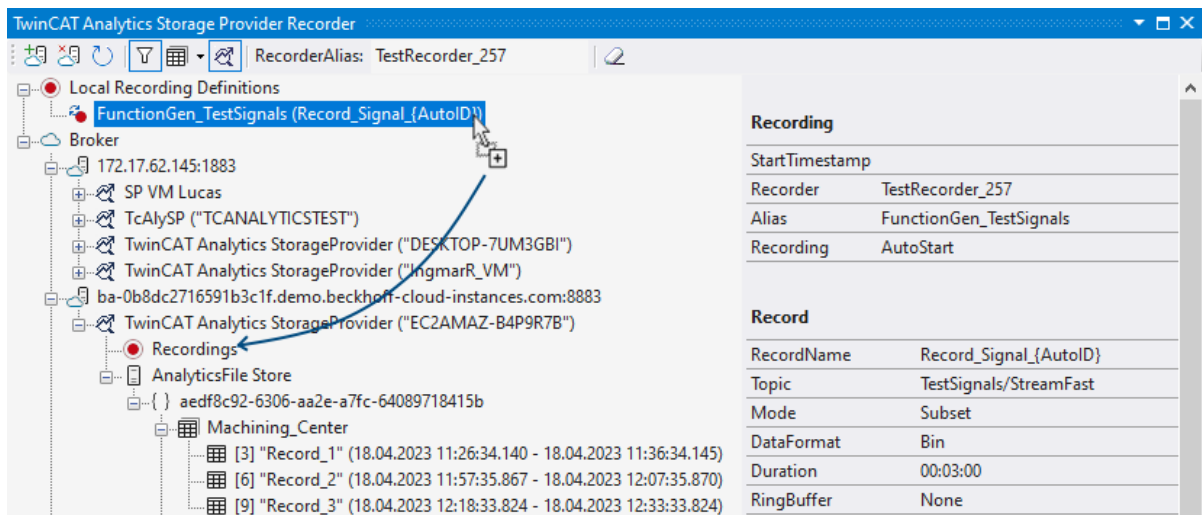
1. Über das Kontextmenü an einem lokalem Aufnahmedefinition Knoten. Hier kann der gewünschte Analytics Storage Provider ausgewählt werden, welcher die Daten aufzeichnen soll. Es werden nur die Storage Provider angezeigt, die auch Zugriff auf das ausgewählte Topic haben.



- Über das Kontextmenü am Analytics Storage Provider Knoten. Hier kann die gewünschte Aufnahmedefinition ausgewählt werden. Es werden nur die Definitionen angezeigt, die auch vom Analytics Storage Provider verarbeitet werden können.

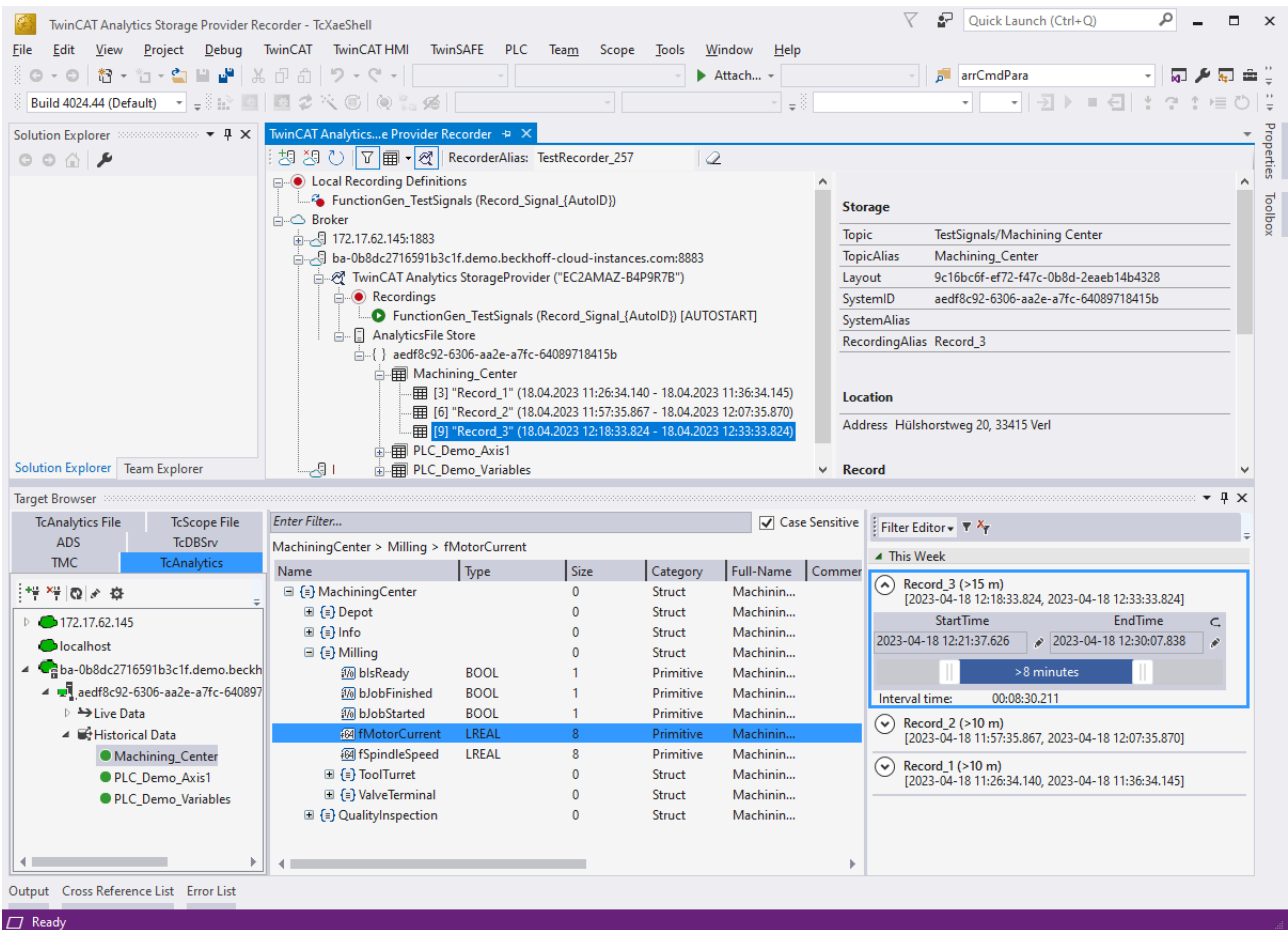


- Per Drag-and-Drop des lokalen Aufnahmedefinition Knoten. Die gewünschte Aufnahmedefinition kann auf den gewünschten Analytics Storage Provider per Drag-and-Drop gestartet werden. Sollte der Storage Provider die Aufnahmedefinition nicht ausführen können, wird eine Fehlermeldung in der Fehlerliste ausgegeben.



**Historisierte Daten verwenden**

Nach und auch während der Aufzeichnung können Sie die historischen Daten als Eingang für Ihre Analyse im Target Browser auswählen. Im Target Browser finden Sie für die historischen Daten eine neue Steuerung auf der rechten Seite. Dort können Sie die Zeitspanne für Ihre Daten auswählen.



## 4.4 Importieren von Analytics Files

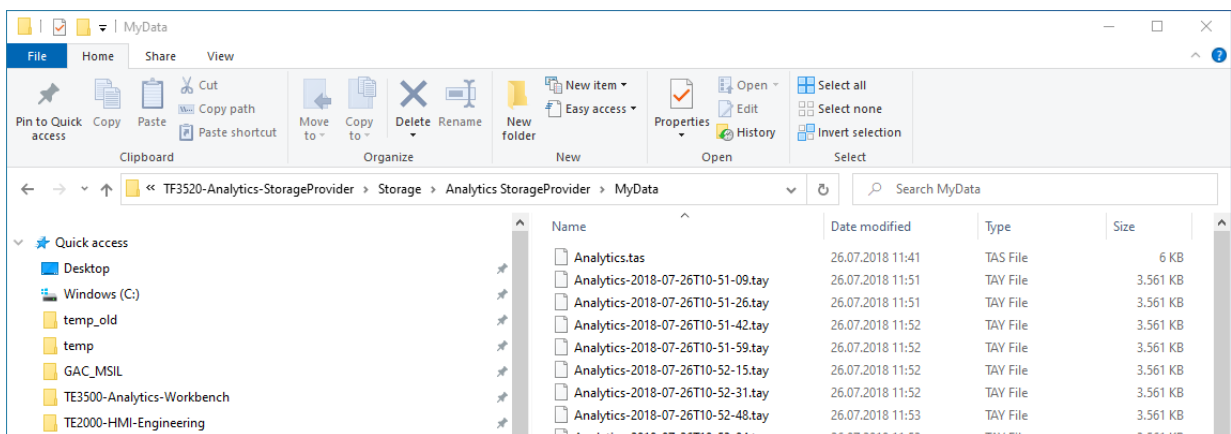


Im Folgenden wird davon ausgegangen, dass Sie TwinCAT unter „C:/TwinCAT“ installiert haben. Andernfalls müssen Sie die angegebenen Pfade entsprechend anpassen.

Sie können Aufnahmen aus dem Analytics Logger, die im Analytics File Format (*Analytics.tas*, *Analytics-<Date>.tay*) gespeichert sind, in den Storage Provider importieren.

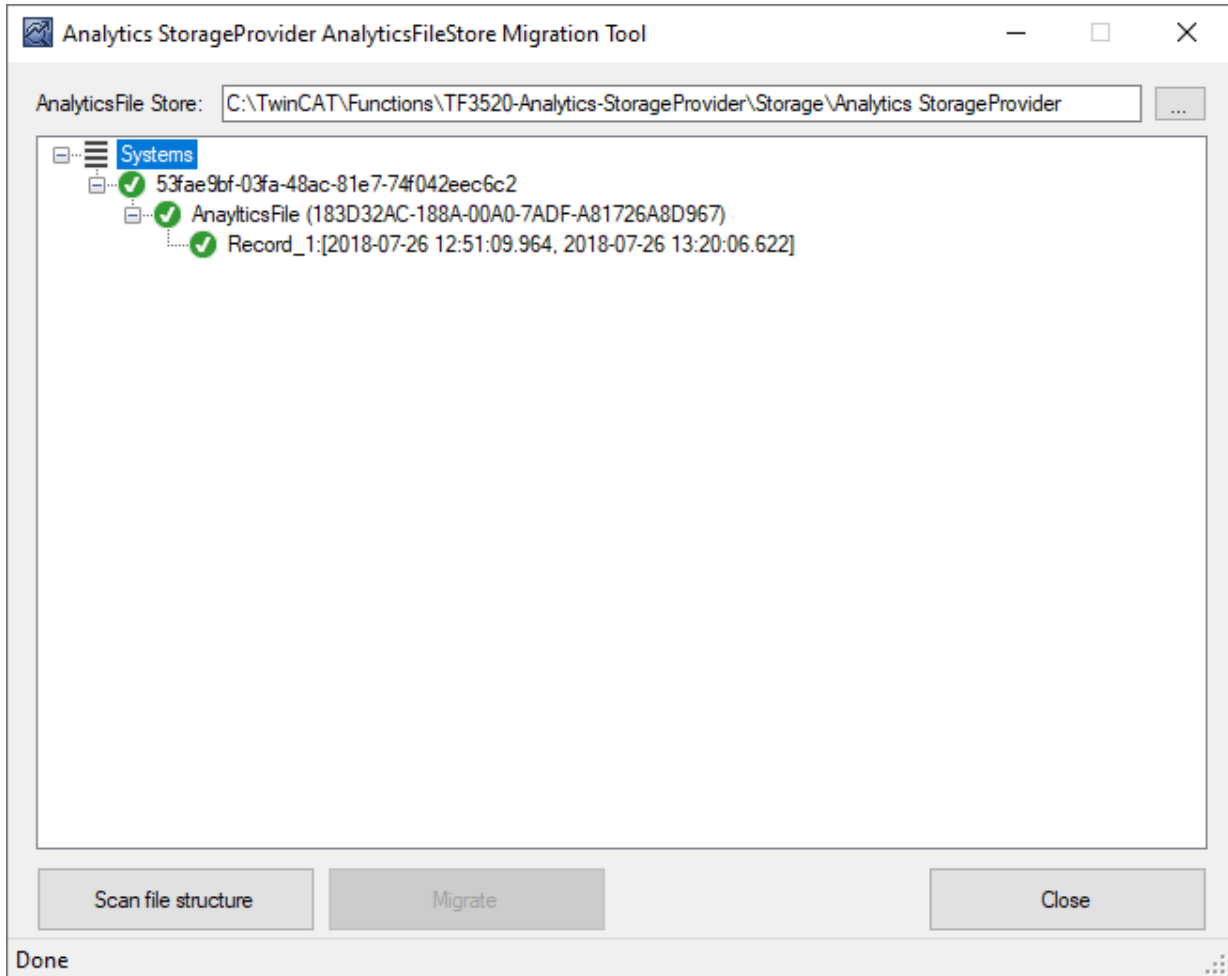
Führen Sie dazu folgende Schritte durch:

1. Legen Sie den Ordner mit den Analytics Dateien in Ihrem Storage Provider Speicherort ab Standardmäßig hier: *C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider\Storage\Analytics-StorageProvider* (legen Sie die Ordner manuell an, falls diese nicht vorhanden sind)



2. Öffnen Sie die *TcAnalyticsSPAlyFileStoreMigration.exe*. Das Programm finden Sie unter dem Pfad *C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider\WinService*.

3. Geben Sie den Pfad zu Ihrem Storage Provider Speicherort an  
Standardmäßig hier: `C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider\Storage`.
4. Drücken Sie *Scan file structure*:
5. Drücken Sie **Migrate**.



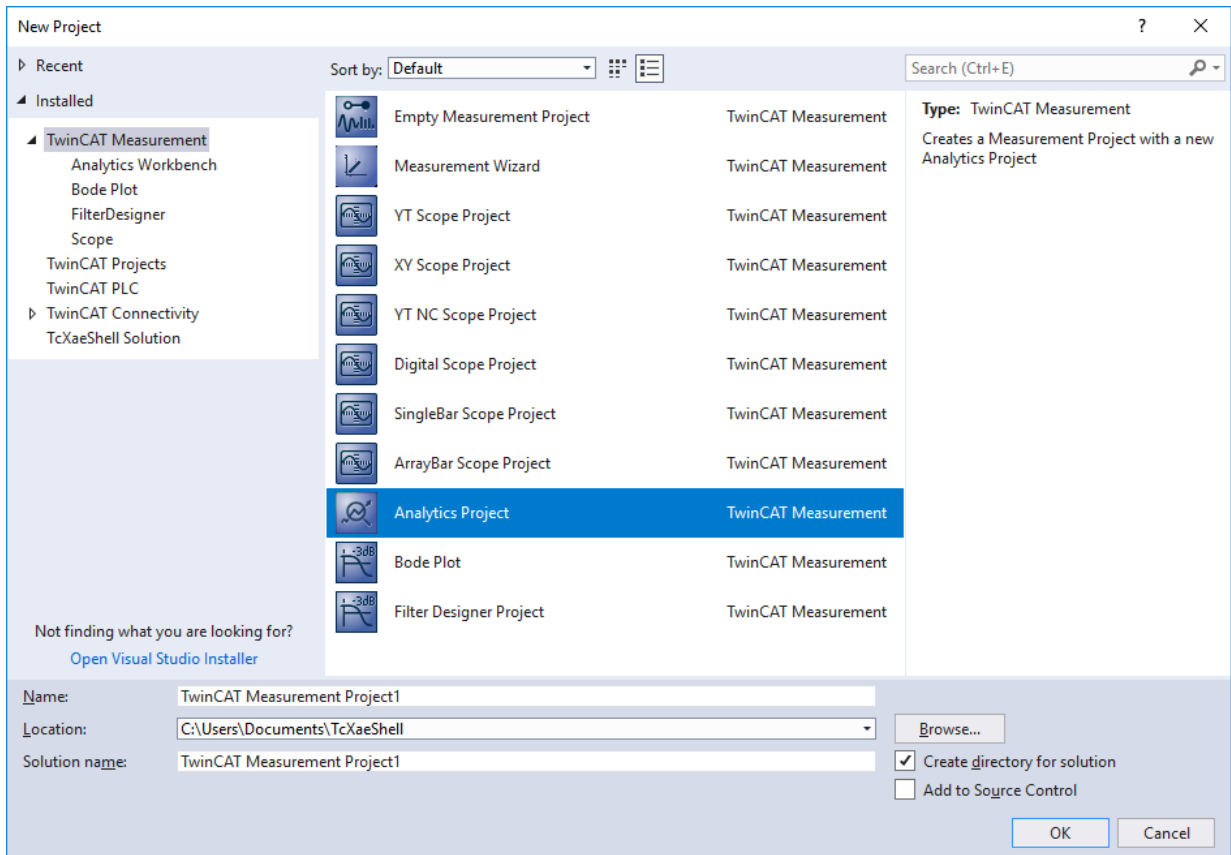
6. Nun können Sie Ihre importierten Daten im TwinCAT Target Browser [► 56] sehen.

Möglicherweise müssen Sie kurz warten bzw. Ihren Storage Provider einmal neu starten.

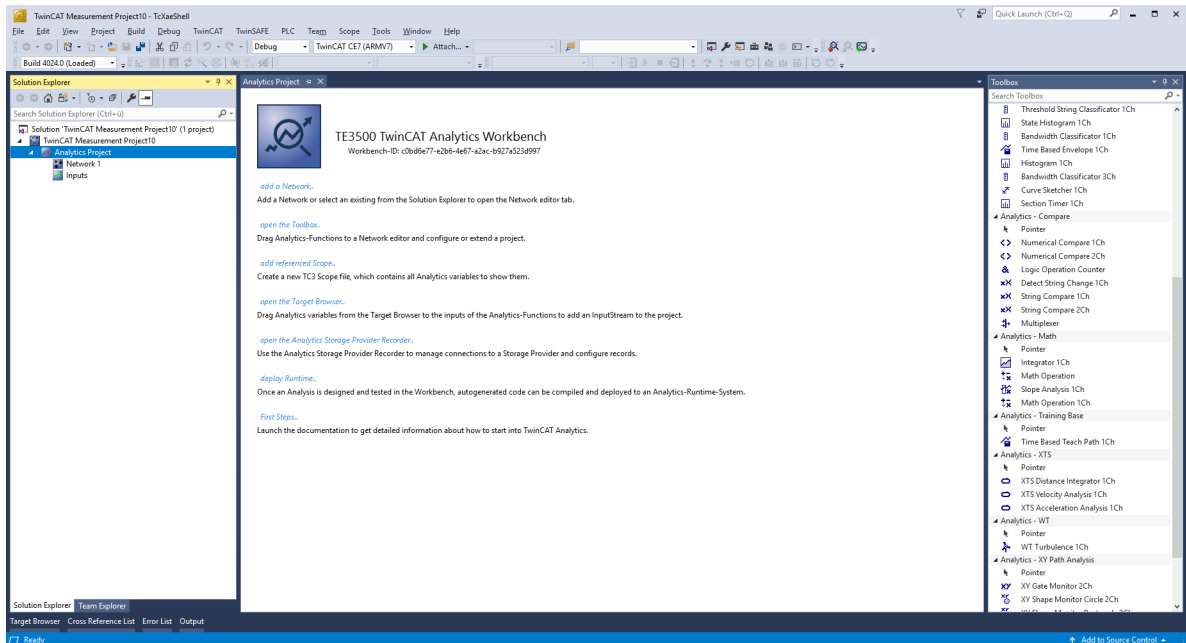
## 4.5 Analyse der Daten

- ✓ Öffnen Sie Ihre TwinCAT Engineering-Umgebung, um die Analyse der Daten zu starten.
1. Öffnen Sie **Visual Studio® > File > New > Project...**

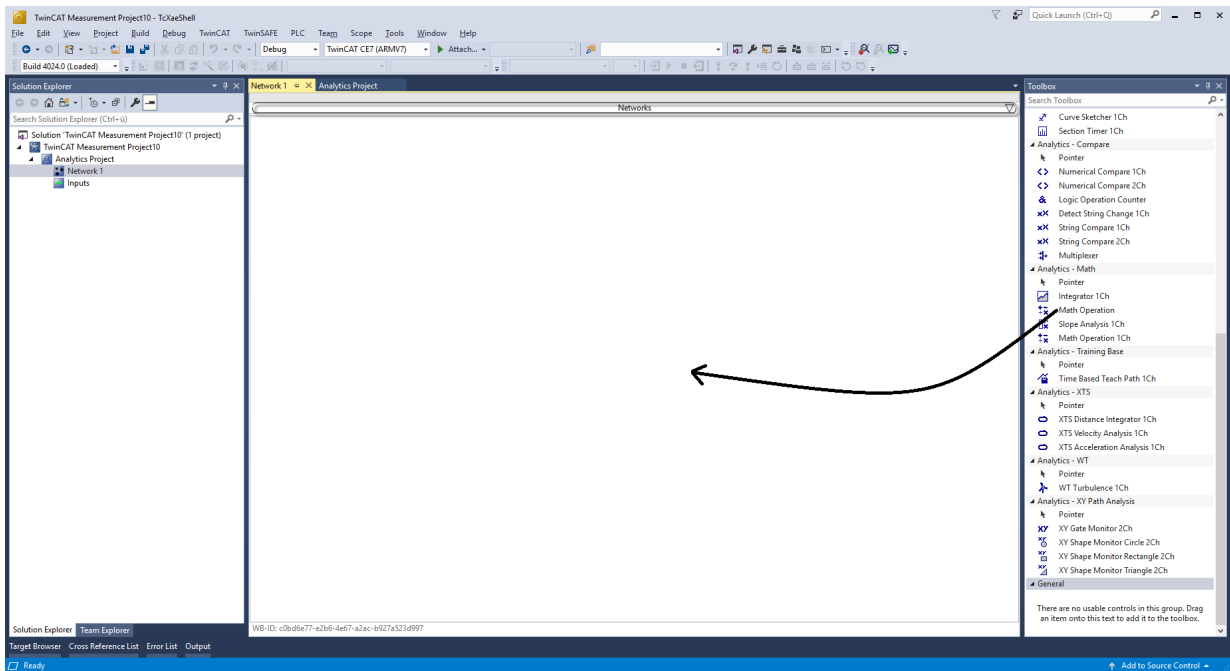
2. Wählen Sie aus **TwinCAT Measurement** die **Analytics-Projektvorlage** aus.



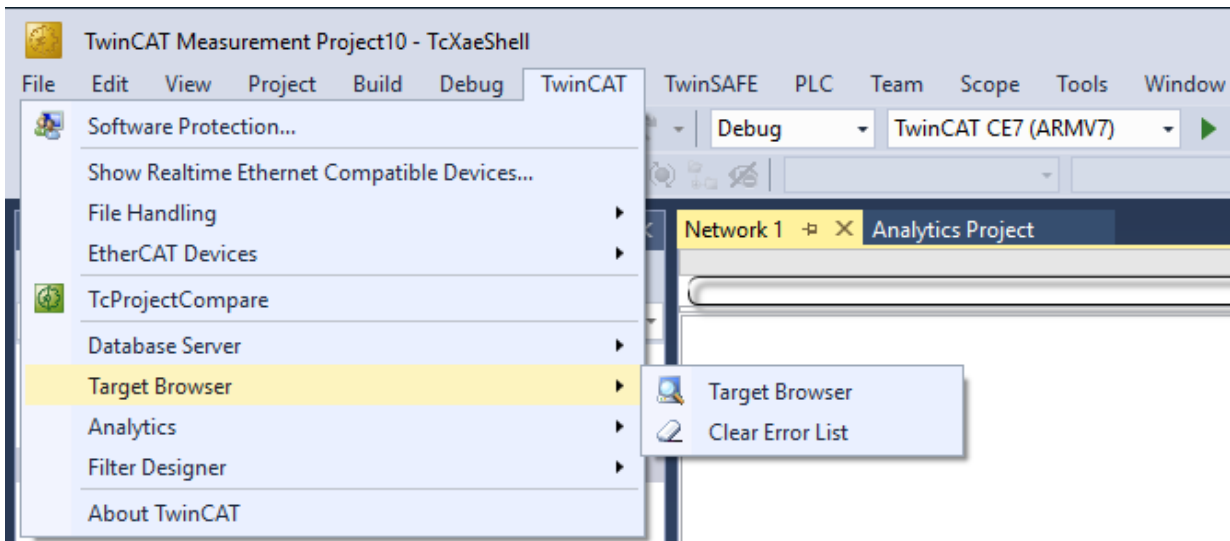
⇒ Das neue Projekt wird im Projektmappen-Explorer angezeigt. Nach einem Klick auf das Baumknotenelement **Analytics Project** öffnet sich ein Startfenster, in dem Sie Ihre erste Aktion auswählen können. Von hier aus können Sie ein Netzwerk hinzufügen, die **Toolbox** öffnen, den **Target Browser** öffnen oder den **Analytics Storage Provider Recorder** öffnen. In den folgenden Schritten werden Sie all diese Aktionen durchführen.



- Es ist sinnvoll, als Erstes die **Toolbox** von Visual Studio® zu öffnen. Dort finden Sie alle unterstützten Algorithmen von TwinCAT Analytics. Die Algorithmen müssen in Netzwerke gruppiert und organisiert werden. Rechtsklicken Sie auf **Analytics Project**, um ein neues Netzwerk hinzuzufügen, oder fügen Sie mit Hilfe der Startseite ein Netzwerk hinzu. Das erste Netzwerk wird immer standardmäßig generiert.

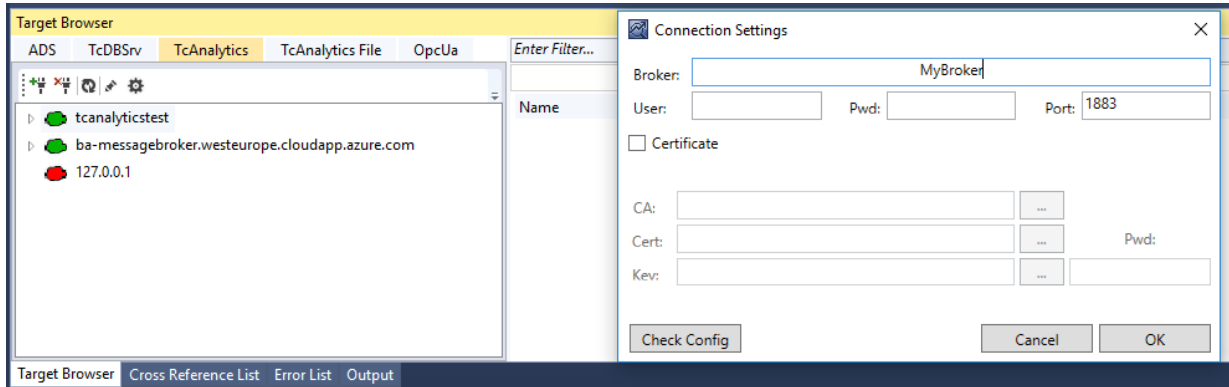


- Wenn Sie auf das Netzwerk klicken, wird ein Editor geöffnet. Nun können Sie den gewünschten Algorithmus per Drag-and-Drop in die Editor-Oberfläche ziehen.
- Nach der Auswahl des Algorithmus müssen Sie Eingangsvariablen mit den Modulen (Algorithmus) verbinden. Öffnen Sie dazu den **Target Browser**.  
**TwinCAT > Target Browser > Target Browser**



- Wählen Sie nun den Karteireiter **TcAnalytics** oder **TcAnalyticsFile** im Target Browser aus. Mit dem Karteireiter **TcAnalytics** (MQTT) geht es weiter.

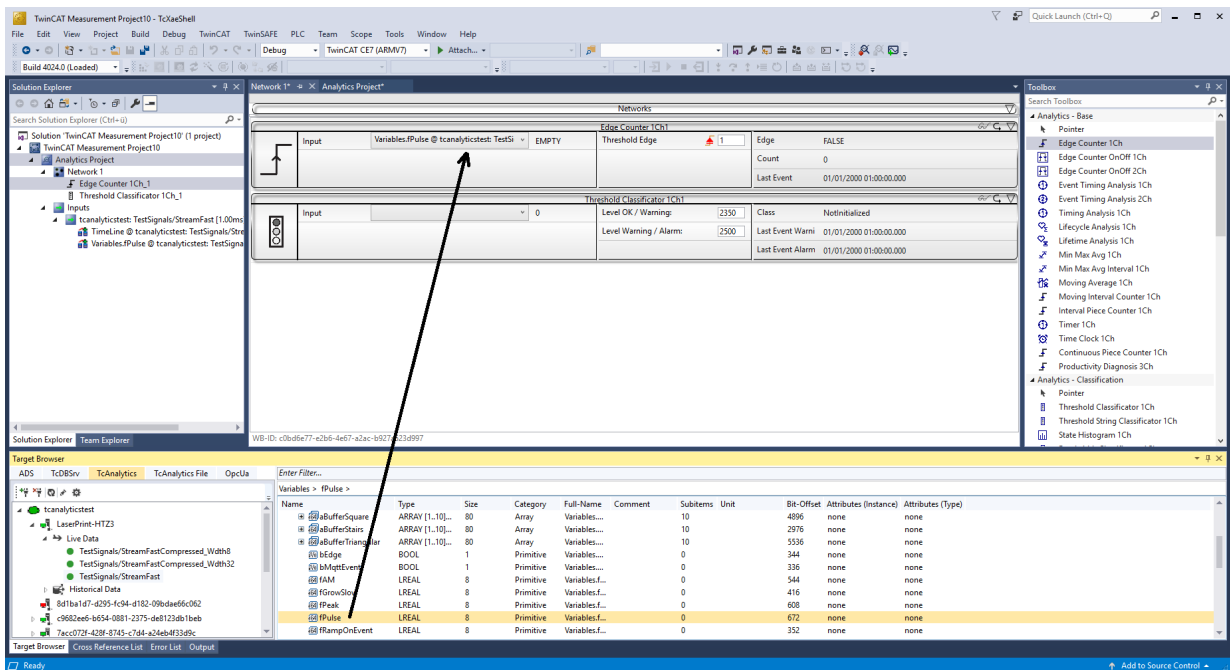
- Klicken Sie auf das grün markierte Icon in der Symbolleiste dieser Analytics-Erweiterung. Es öffnet sich ein Fenster, in dem Sie die Konnektivitätsdaten Ihres Message-Brokers angeben können.



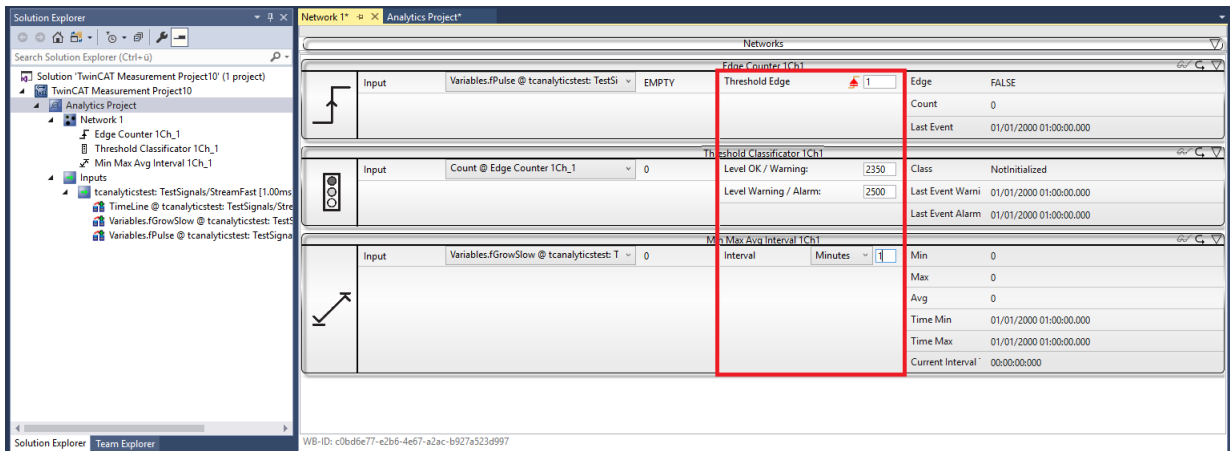
- Wählen Sie Ihren MQTT-Analytics-Client aus (TwinCAT Analytics Logger, TwinCAT IoT Data Agent oder Beckhoff EK9160). Für jede Steuerung gibt es eine eindeutige ID. Diese ID wird im Target Browser angezeigt.
- Über einen Klick auf das **Zahnrad-Icon**, gelangen Sie zur Seite Machine Administration. Hier können Sie einen System Alias-Namen vergeben, der im Target Browser anstelle der ID angezeigt wird.

Source	Customer	SystemID	System Alias	Online	Position	Topic Alias	Description
172.17.62.145							
TwinCAT Analytics Logger	MA Laser	3db95703-29fb-d99e-eb13-017b54677bb0	LaserPrintZG15	True			
TwinCAT Analytics Logger	MA Laser	3db95703-29fb-d99e-eb13-017b54677bb0	LaserPrintZG15	True	Laserstr. 13, 40597 Düsseldorf		MP4 Steel Laser Data
TwinCAT Analytics Logger	MA Laser	3db95703-29fb-d99e-eb13-017b54677bb0	LaserPrintZG15	True			
TwinCAT Analytics Logger	Fertig Motors	8d1ba1d7-d295-fc94-d182-09bdae66c062		False			
TwinCAT Analytics Logger	Fertig Motors	8d1ba1d7-d295-fc94-d182-09bdae66c062		False			
TwinCAT Analytics PLC DataLogger		7acc072f-428f-8745-c7d4-a24eb4f33d9c		False			
TwinCAT Analytics TEST Logger	Beckhoff	91c6eab3-1abb-5593-3651-1426874c91f		True	Hülshorstweg 20, 33415 Verl		
TwinCAT Analytics Storage Provider		c9682ee6-b654-0881-2375-de8123db1beb		True		RetroFit	
TwinCAT Analytics Storage Provider		7acc072f-428f-8745-c7d4-a24eb4f33d9c		True		AnalyticsSolution Results 1	
TwinCAT Analytics Storage Provider		7fb4f250-c130-7d7e-0a26-71ed8cee9340		True		CloudControl	
TwinCAT Analytics Storage Provider		3db95703-29fb-d99e-eb13-017b54677bb0	LaserPrintZG15	True			
TwinCAT Analytics Storage Provider		3db95703-29fb-d99e-eb13-017b54677bb0	LaserPrintZG15	True		TestSignals whole Image	
TwinCAT Analytics Storage Provider		c5eefcfd-4f14-5f45-dce4-7524715a9ae3		True		DataAgent Test	
TwinCAT Analytics Storage Provider		3f8a342a-6fac-3e76-6172-e7b5f62c0eb0		True		BigI40 FavValues	
TwinCAT Analytics Storage Provider		a313c550-7537-0617-827d-c6930e90d931		True		EK Test2	
TwinCAT Analytics Storage Provider		d180dde-afea-78d2-9ac1-65101d008687		True		NewMachineApp	
TwinCAT Analytics Storage Provider		3db95703-29fb-d99e-eb13-017b54677bb0	LaserPrintZG15	True		LongTerm	
TwinCAT Analytics Storage Provider		3db95703-29fb-d99e-eb13-017b54677bb0	LaserPrintZG15	True		MyFavoriteData	
TwinCAT Analytics Storage Provider		56cfbec6-3ab5-c1cc-1a1d-e6f4da86adf0		True		EdgeComputingTc2	
TwinCAT Analytics Storage Provider		56cfbec6-3ab5-c1cc-1a1d-e6f4da86adf0		True		EdgeComputingTc3	

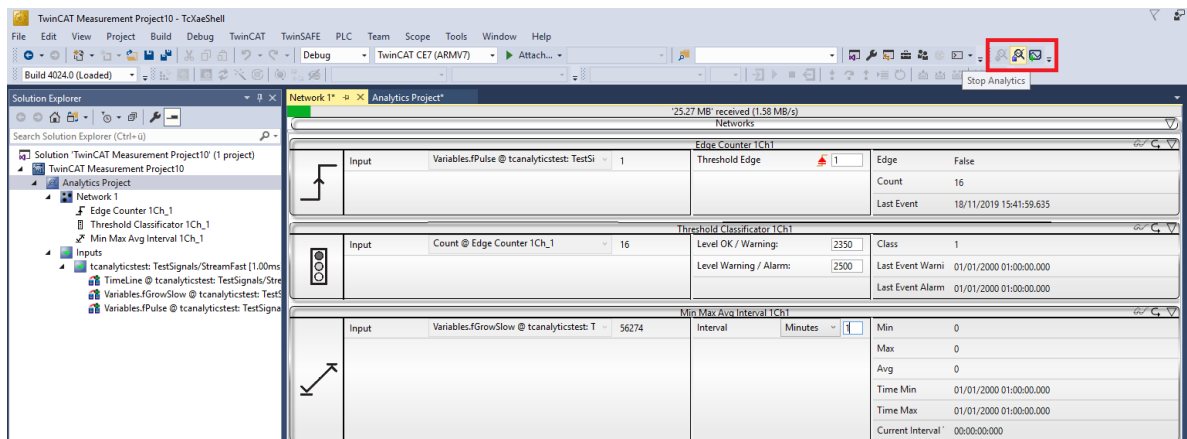
10. Im nächsten Schritt können Sie zwischen Live-Daten und historischen Daten für jeden MQTT Analytics-Client wählen. Die historischen Daten werden in diesem Fall von dem TwinCAT Analytics Storage Provider bereitgestellt.



11. Sie können die Variablen per Drag-and-Drop in die Eingänge des spezifischen Algorithmus ziehen. Bei den meisten Algorithmen können Bedingungen wie Schwellen, Zeitintervalle, logische Operatoren usw. festgelegt werden. Diese Einstellungen werden in der Mitte jedes Moduls vorgenommen.

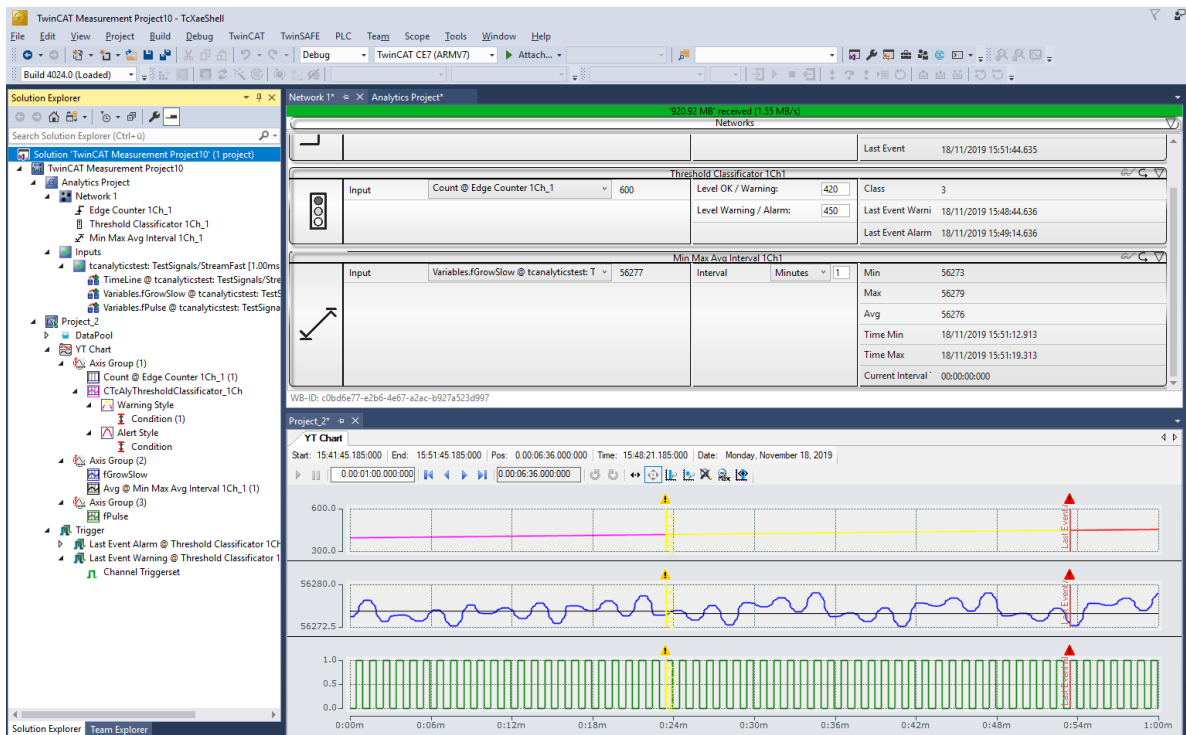


⇒ Schließlich ist Ihr erstes Analytics Project fertiggestellt. Um die Analyse zu starten, klicken Sie auf **Start Analytics**. Um die Analyse zu stoppen, klicken Sie auf **Stop Analytics**.





- ⇒ Vor dem Start der Analyse oder auch während der Laufzeit können Sie auf die Schaltfläche **Add Reference Scope** klicken. Dadurch wird automatisch eine zu Ihrem Analytics-Projekt passende Scope-Konfiguration erstellt.

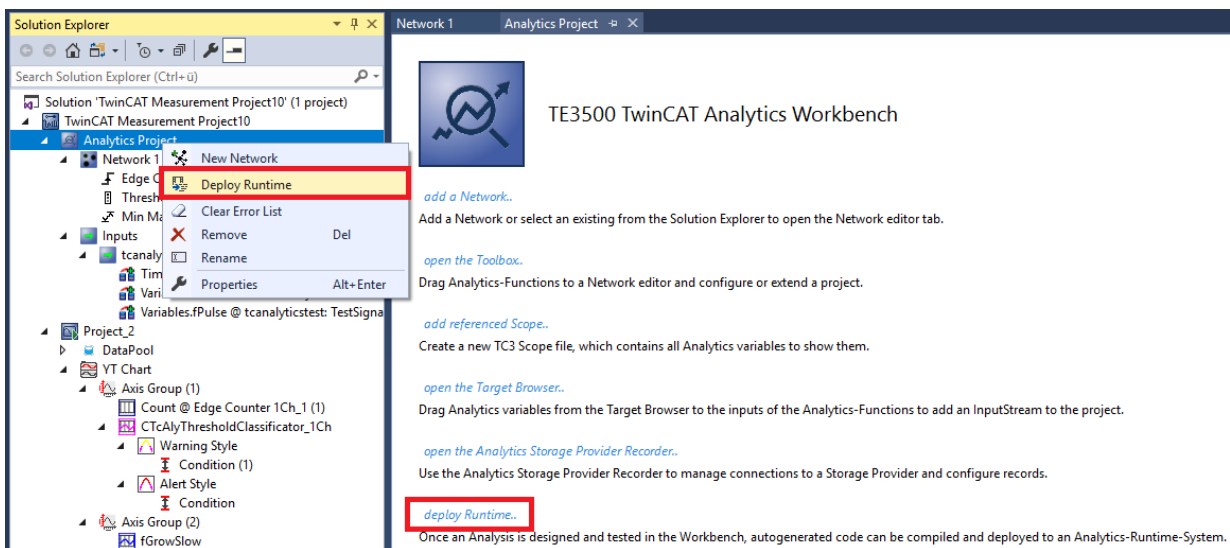


- ⇒ Die Ergebnisse der Analyse können in den Scope View-Grafiken per Drag-and-Drop angezeigt werden. Beispielsweise kann ein Mittelwert als neuer Kanal in der Ansicht angezeigt werden. Zeitstempel als Marker auf den X-Achsen zeigen signifikante Werte.

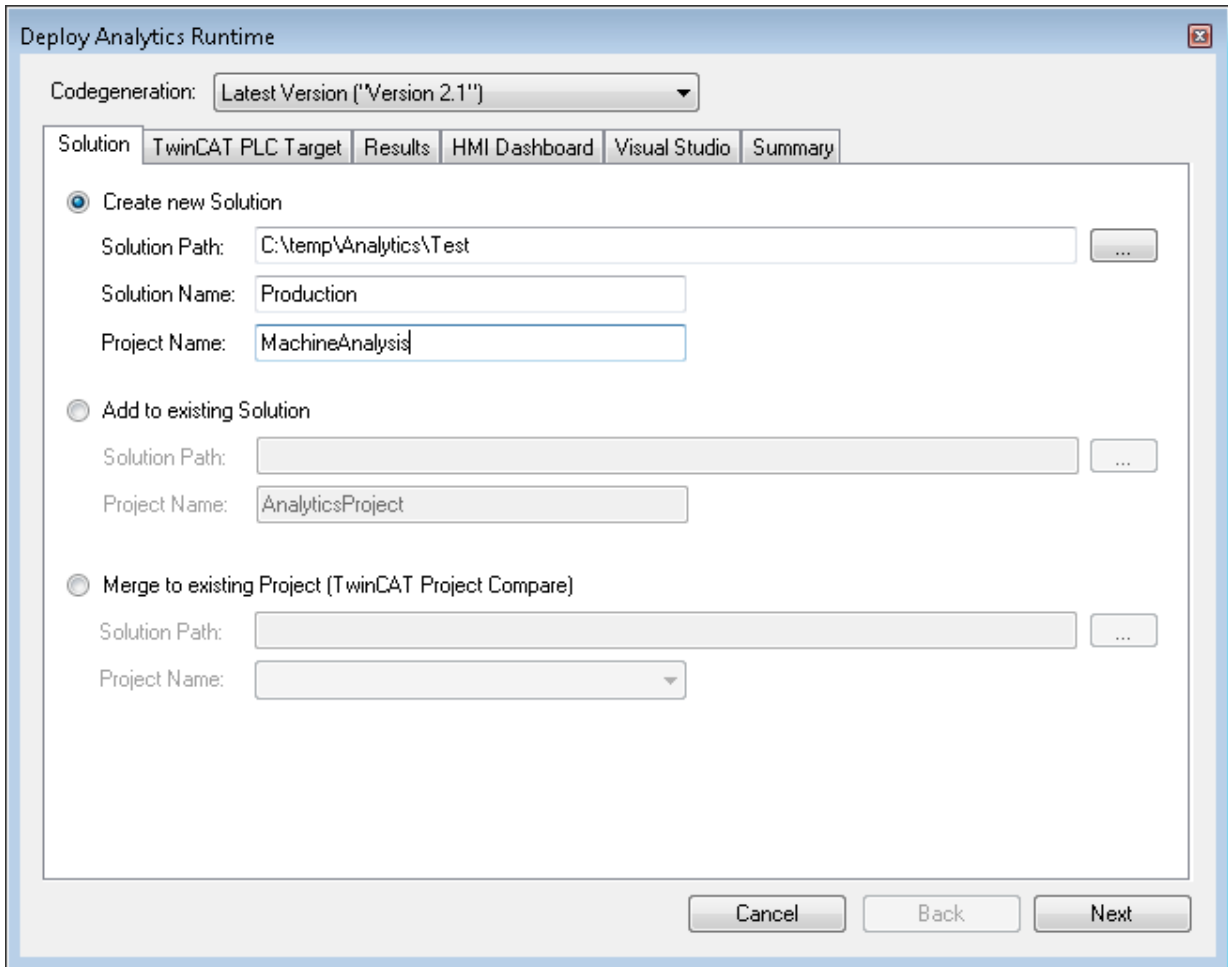
## 4.6 24-stündige Anwendung von Analytics

Der letzte große Schritt im Workflow von TwinCAT Analytics ist die kontinuierliche 24-stündige Maschinenanalyse. Sie läuft parallel zu den Maschinenanwendungen im Feld. Um dies ganz einfach zu gestalten, kann die TwinCAT Analytics Workbench automatisch einen SPS-Code und ein HTML5-basiertes Dashboard Ihrer Analytics-Konfiguration generieren. Beide können in eine TwinCAT Analytics Runtime (TC3 PLC und HMI Server) heruntergeladen werden und bieten dieselben Analyseergebnisse wie das Konfigurator-Tool in der Engineering-Umgebung.

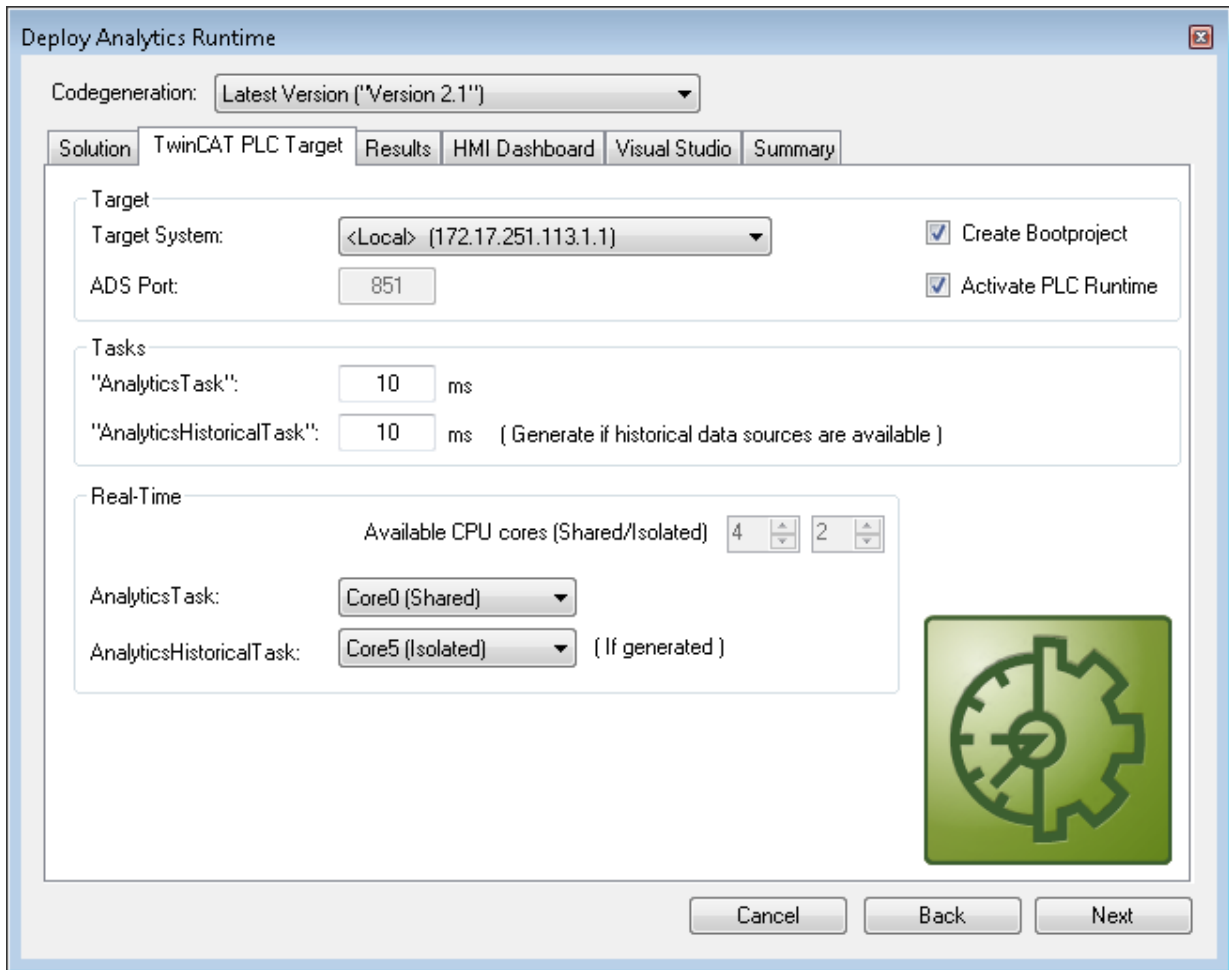
- ✓ Speichern Sie zunächst Ihre Konfiguration und öffnen Sie den Analytics Deploy Runtime-Assistenten. Dies kann über das Kontextmenü im Analytics Project-Tree Item oder über die Startseite erfolgen.



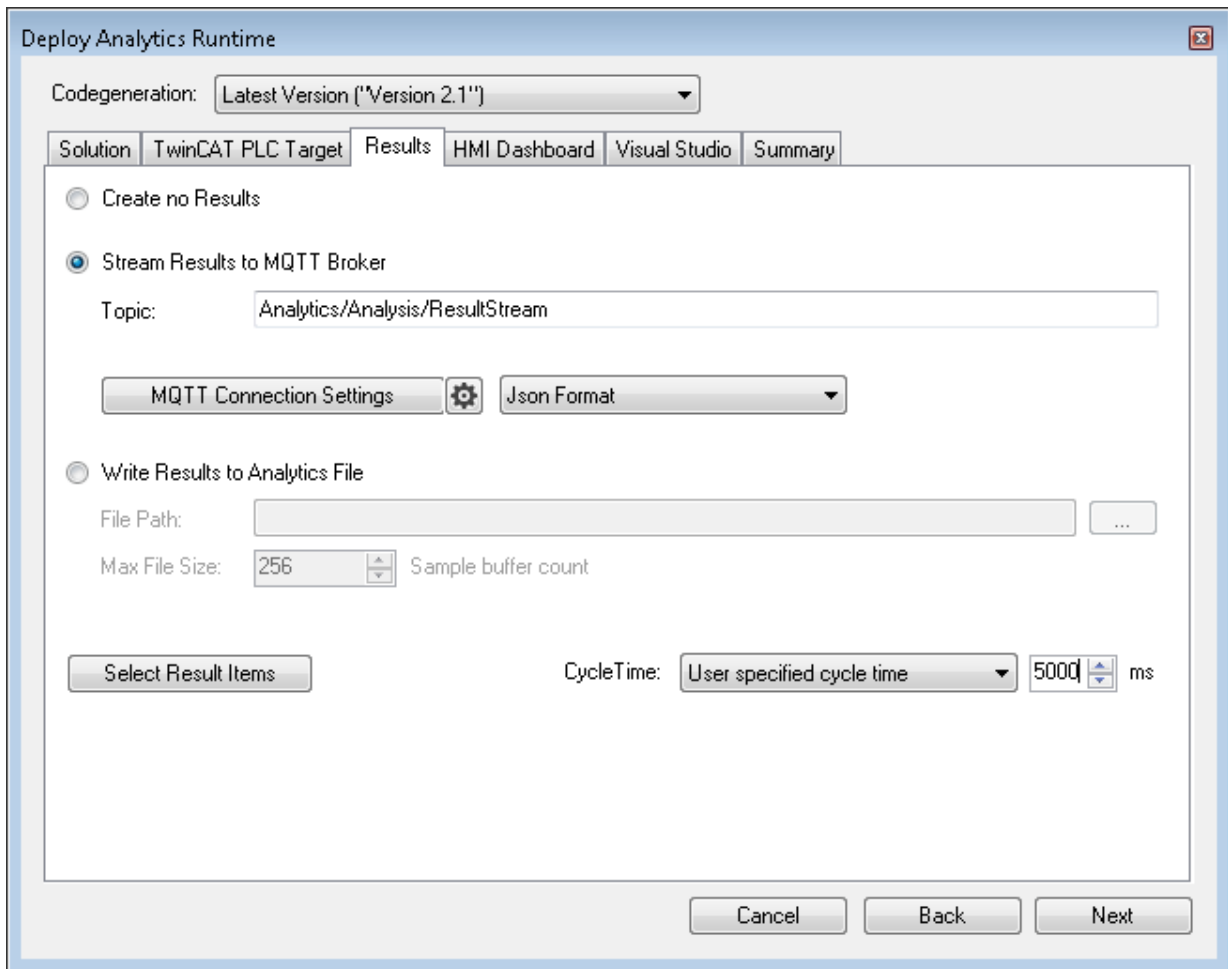
1. Wenn der Assistent geöffnet ist, können Sie sich durch einige Karteireiter klicken. Der erste heißt Solution. Hier können Sie entscheiden, wie Ihr Analytics-Projekt im SPS-Code verwendet werden soll:  
 Als...  
 vollständig neue Lösung.  
 Teil einer vorhandenen Lösung.  
 Aktualisierung einer vorhandenen Analytics-Lösung.



- Im Karteireiter **TwinCAT PLC Target** können Sie das ADS-Zielsystem wählen, das die TwinCAT Analytics Runtime (TF3550) ausführt. Das erzeugte Projekt ist sofort ausführbar. Hierzu können Sie die Option **Activate PLC Runtime** setzen. Zudem kann ausgewählt werden, dass direkt ein Bootprojekt erstellt wird.

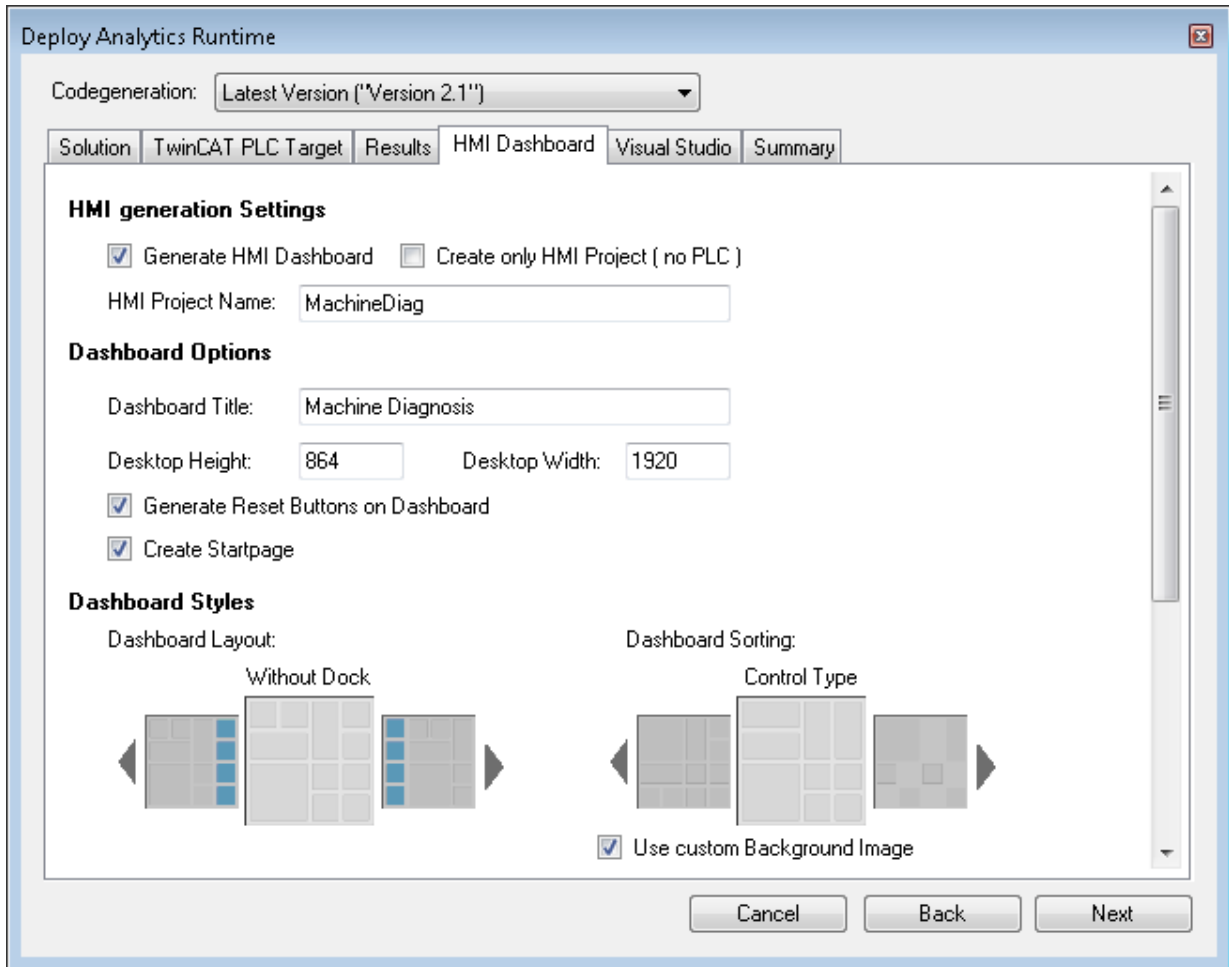


- Insbesondere bei virtuellen Maschinen ist es wichtig, das Projekt auf isolierten Kernen auszuführen, was ebenfalls eine Option in diesem Karteireiter ist. Der nächste Karteireiter **Results** wird nur benötigt, wenn Sie in den Algorithmeigenschaften die Option **Stream Results** ausgewählt haben. Wenn Sie Ergebnisse senden möchten, können Sie hier entscheiden, in welcher Weise (lokal in einer Datei/durch MQTT) und welchem Format (binär/JSON) dies geschehen soll. Auch dies wird automatisch generiert und umgehend nach Aktivierung ausgeführt.

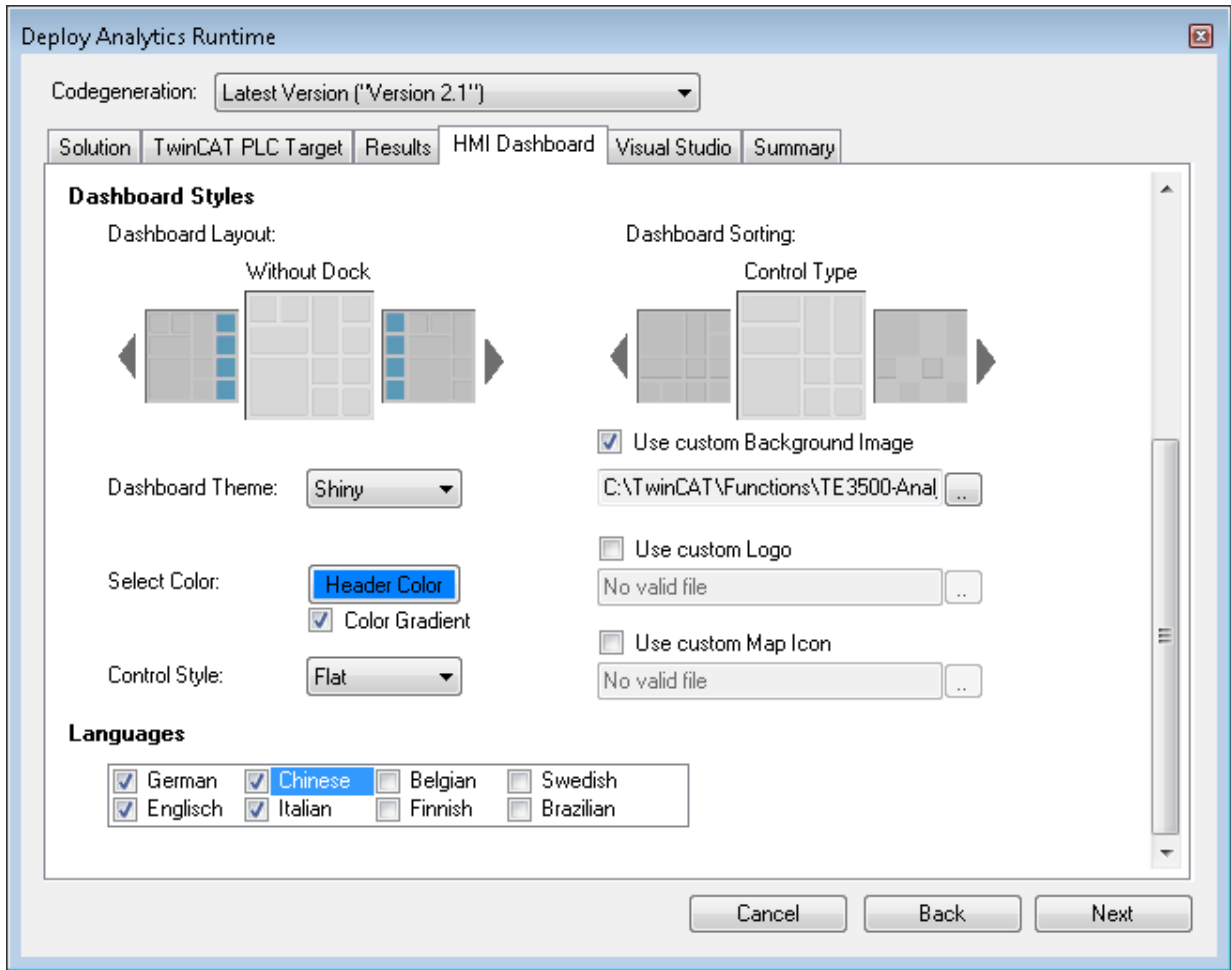


Ein Downsampling der Ergebnisse ist durch die Festlegung einer Zykluszeit möglich. Der nächste Karteneiter ist für das **HMI-Dashboard** vorgesehen. Voraussetzung für die automatische Generierung

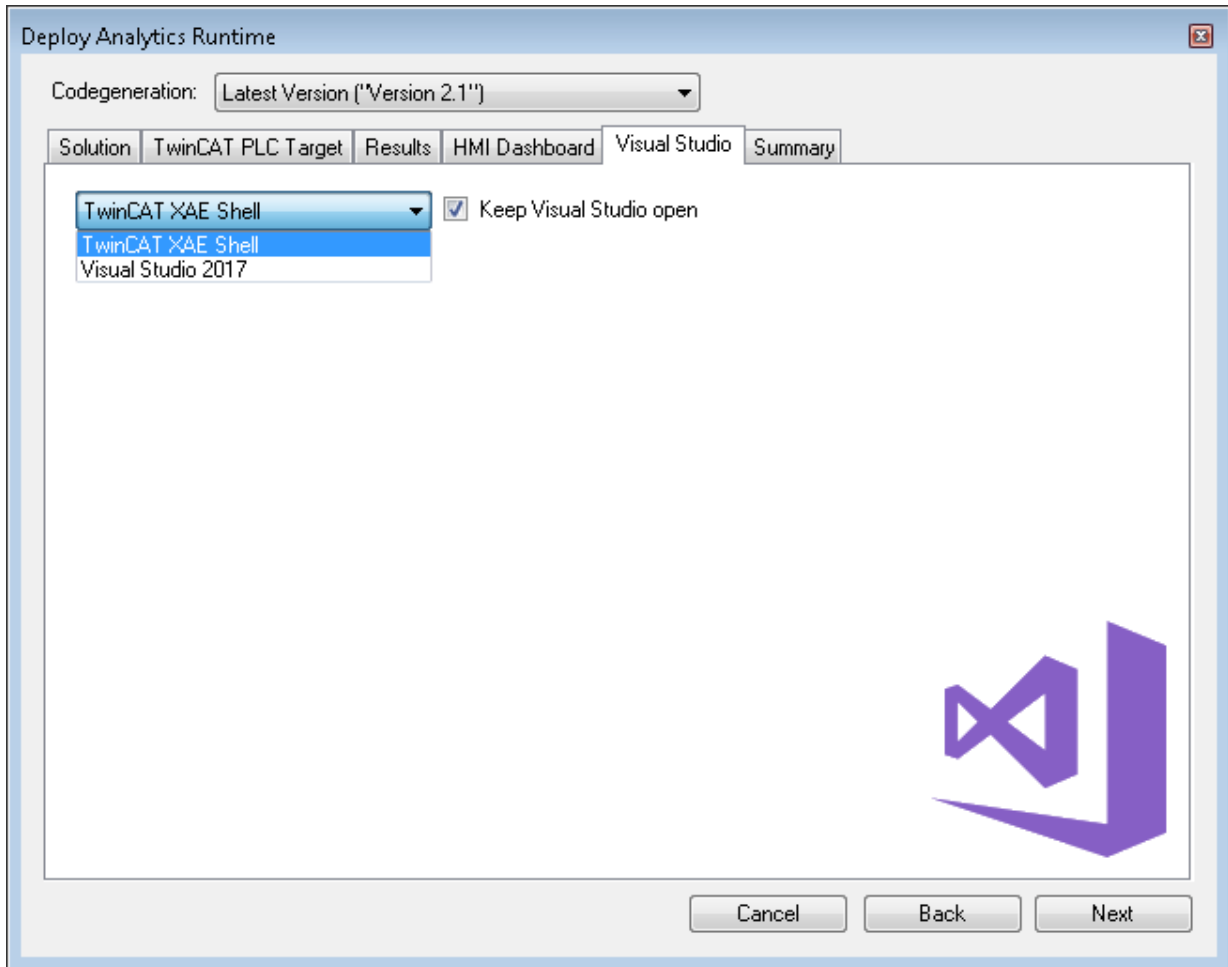
des Dashboards ist die Auswahl von HMI Controls für die entsprechenden Algorithmen, deren Ergebnisse im Dashboard angezeigt werden sollen.



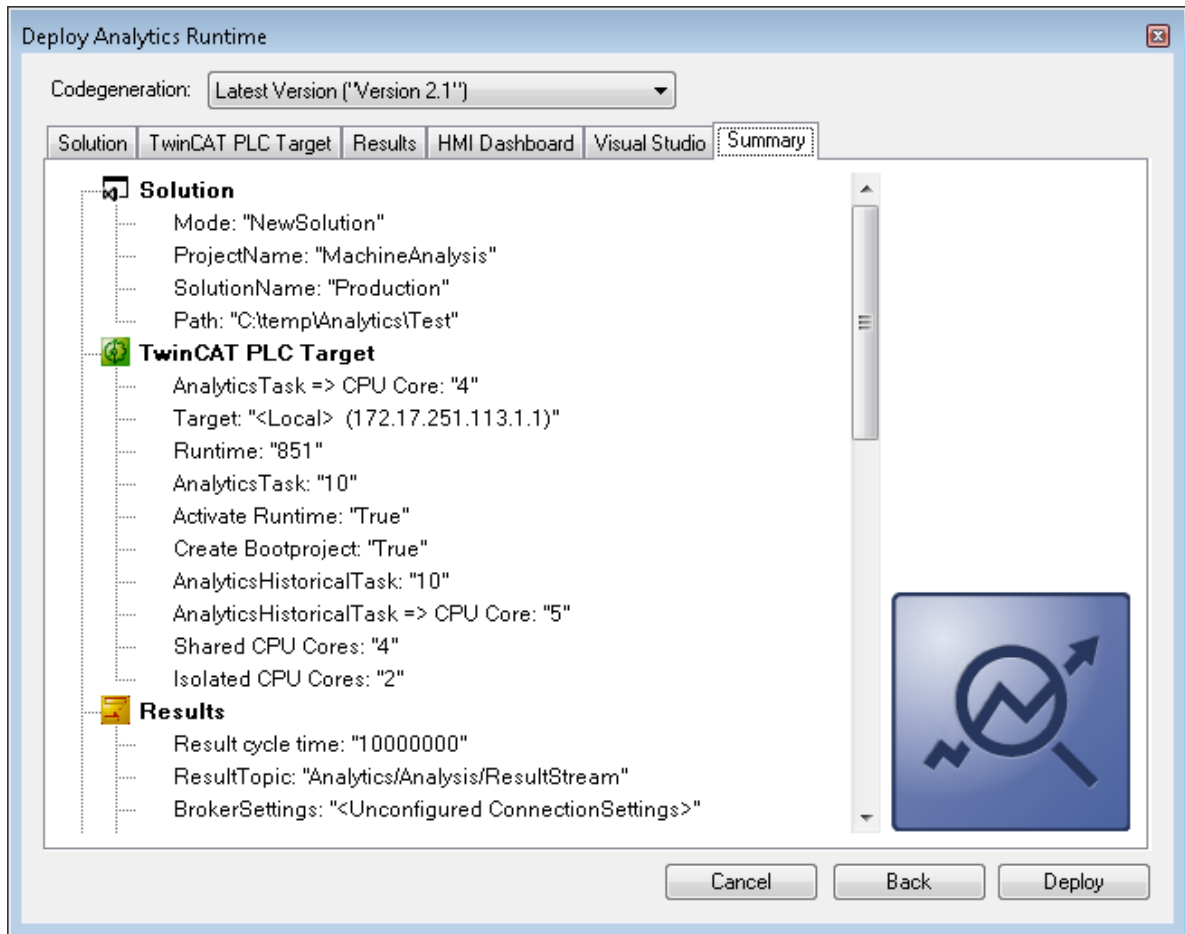
4. Sie können verschiedene Optionen für Ihr Analytics Dashboard auswählen, wie eine Startseite mit einer Karte, Layouts, Sortieralgorithmen, eigene Farben und Logos. Wenn Sie mehrere Sprachen für die Analytics Controls auswählen, wird auch ein Menü zur Sprachumschaltung generiert.



5. Wählen Sie eine der installierten Versionen von Visual Studio® aus, und, ob die Instanz sichtbar starten soll oder ob sie nur eingerichtet und im Hintergrund aktiviert werden soll.

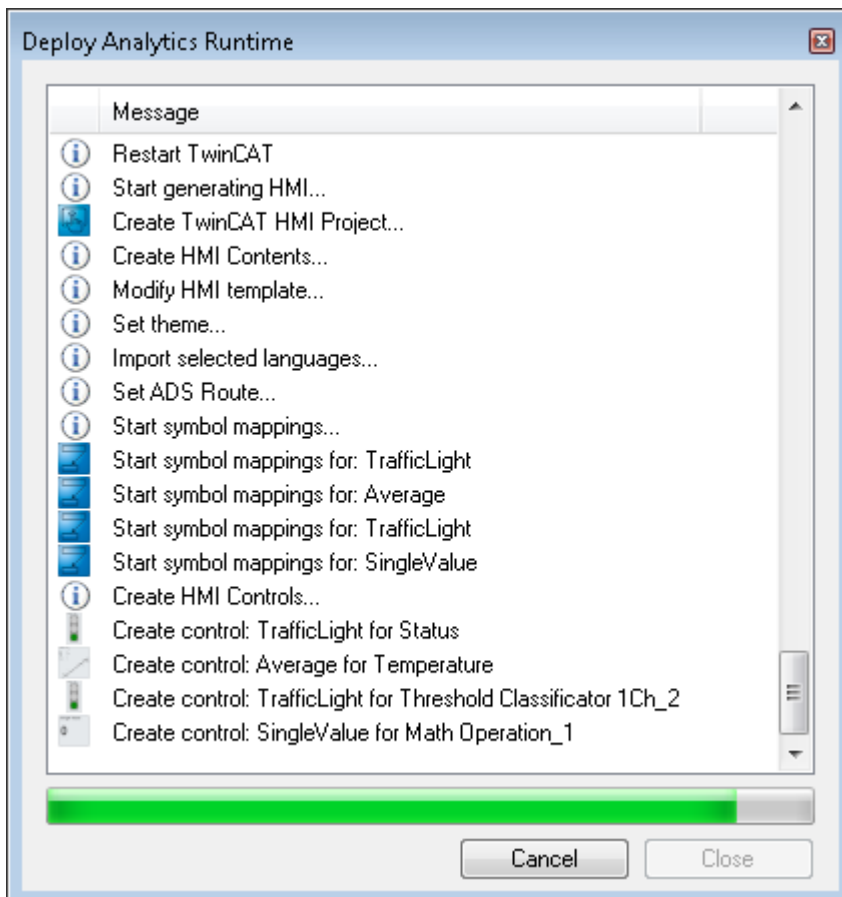
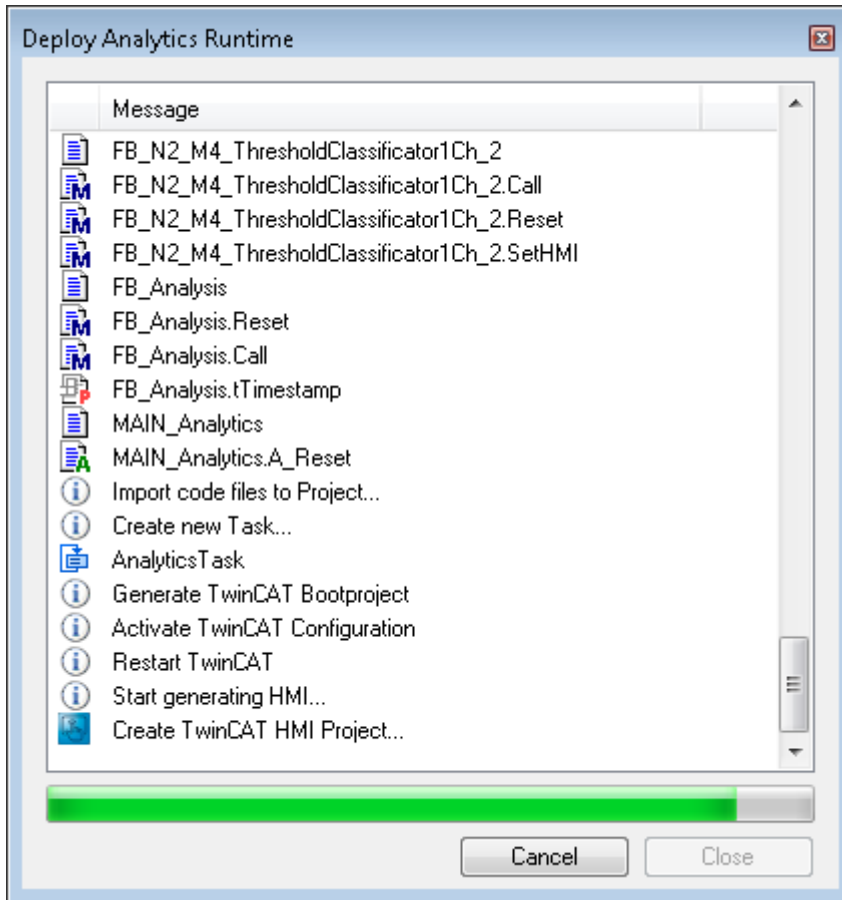


⇒ Zuletzt finden Sie eine Übersicht.

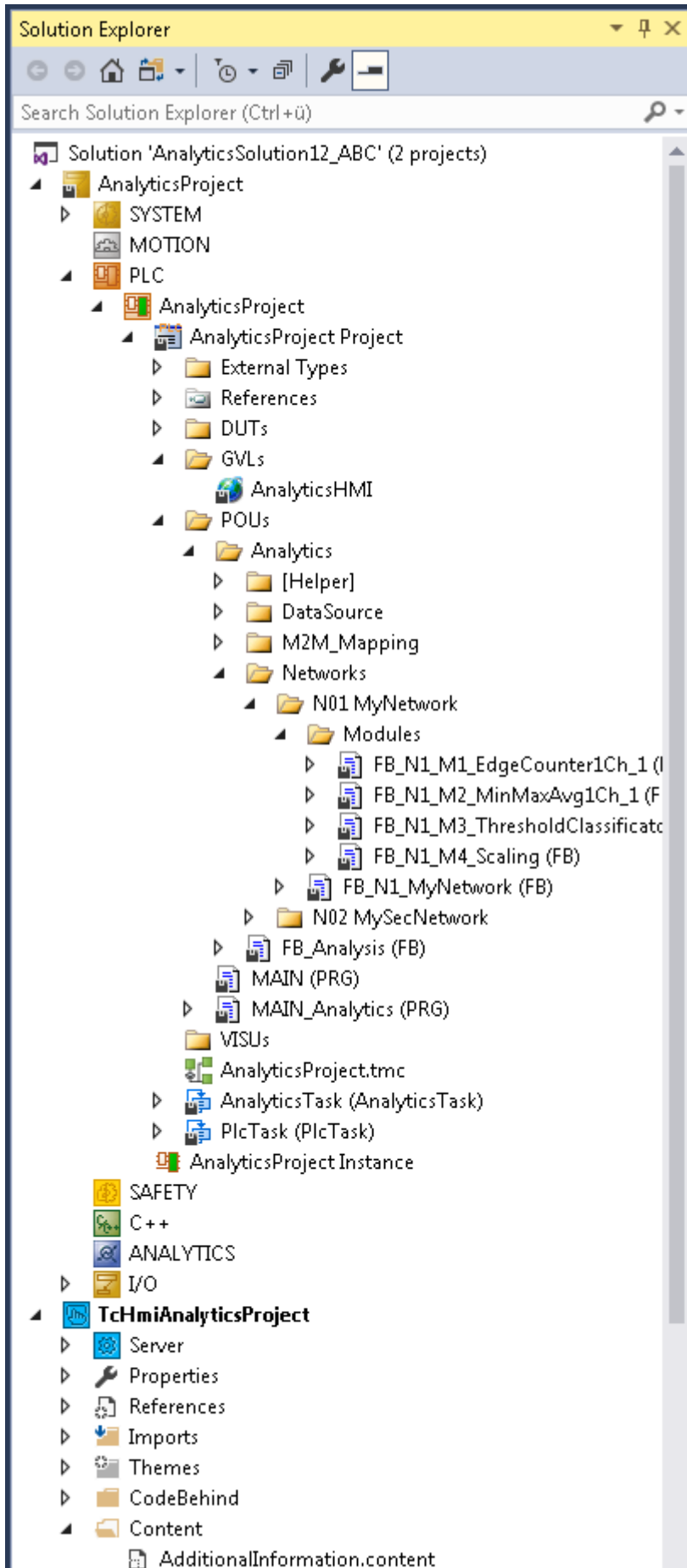




- Nun können Sie auf die Schaltfläche **Deploy** klicken, um den Generierungsprozess zu starten. Das SPS-Projekt und das HMI-Dashboard werden nun generiert.



⇒ Nach der Meldung „Deploy Runtime succeeded“ finden Sie eine neue Visual Studio®/XAE Shell-Instanz auf Ihrem Desktop. Die neue Projektmappe und beide Projekte werden erstellt.



## 5 Technische Einführung

Die Grundidee des TwinCAT Analytics Storage Providers (ASP) besteht darin, ein Gateway zu haben, das den Benutzer von der Konfiguration einer Datensenke, also einem Storage bzw einer Datenbank, weitgehend befreit. Der Benutzer muss keine eigene Tabellenstruktur in einer Datenbank einrichten. Er muss lediglich konfigurieren, welche der unterstützten Datensenken er für die Ablage seiner Daten nutzen möchte.

### Service-Management

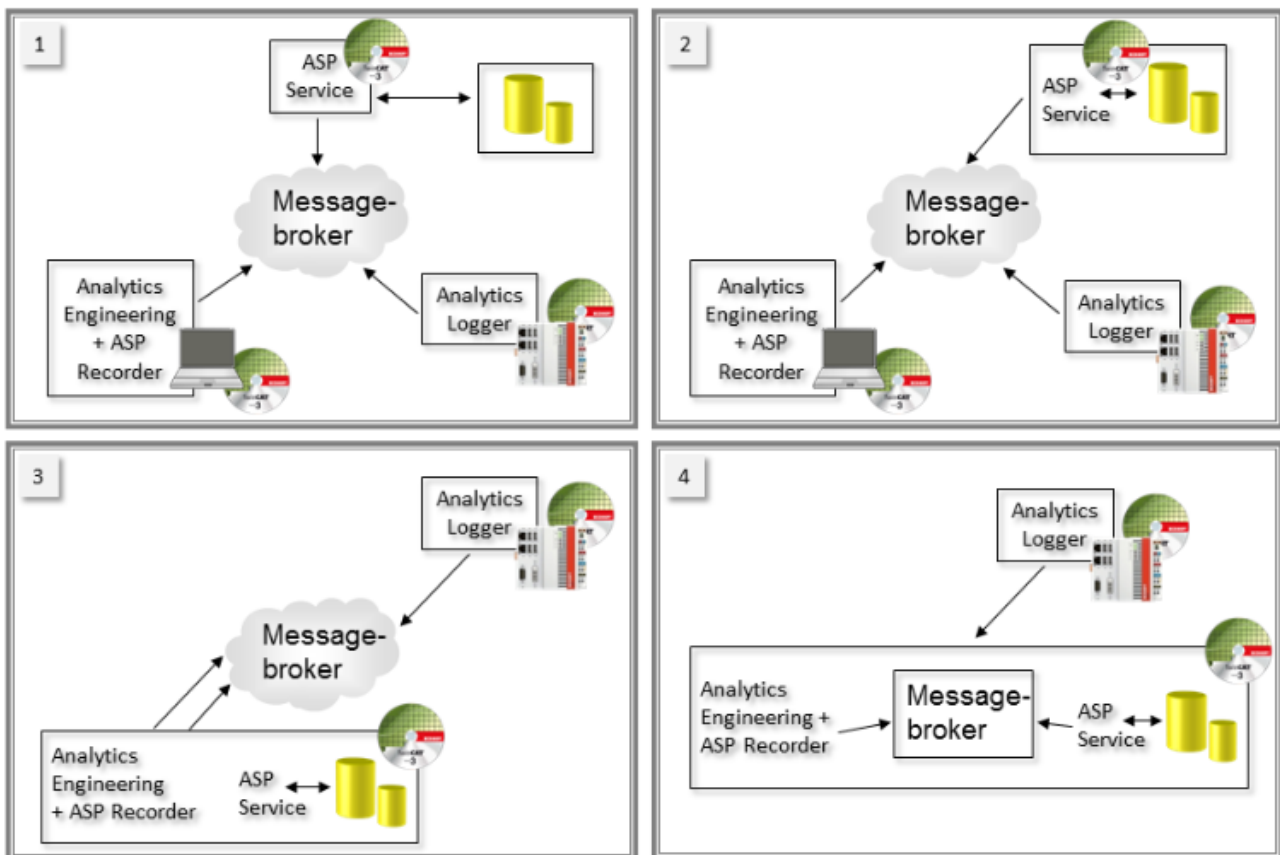
Der Analytics Storage Provider Service kann irgendwo im Netzwerk ausgeführt werden. Er ist als Windows Service implementiert. Der Dienst kann auf Hardware-Geräten, wie Industrie-PCs oder Embedded-PCs im lokalen Netzwerk, und auch auf virtuellen Maschinen im selben Netzwerk, oder z. B. in einem Cloud-System ausgeführt werden.

### Daten-Management

Der Storage Provider arbeitet mit dem Binärformat von TwinCAT Analytics. Dadurch kann er Streams von einem MQTT-Message-Broker empfangen und speichern sowie selbst neue Streams erstellen und senden. Der Benutzer benötigt dazu lediglich den Recorder, der mit der TwinCAT Analytics Workbench oder dem Service Tool in seinem eigenen Engineering-System integriert wird. Die Variablen selbst werden im TwinCAT Target Browser angezeigt. Für das Analytics-Binärformat werden sie in Live- und historische Daten unterteilt. Live-Daten können als Eingang für den Analytics Storage Provider verwendet werden. Historische Daten sind die Werte aus der Datenbank/dem Storage, die bzw. der vom Storage Provider bereitgestellt wird.

### Topologien

Die vielen Freiheitsgrade der IoT-Technologien ermöglichen verschiedene Topologien. Das folgende Bild zeigt die wichtigsten Konstellationen.



1. Jedes SW-Paket läuft auf einem eigenen HW-Gerät oder einer virtuellen Maschine.
2. Der Windows Service des Analytics Storage Providers läuft auf demselben Gerät wie die Datenbank/ der Storage.

3. Analytics Engineering, Analytics Storage Provider und Datenbank oder Storage befinden sich auf demselben Gerät. Nur der Message-Broker und der Analytics Logger (Datenquelle) laufen auf anderen Geräten.
4. In dieser Topologieansicht läuft nur der Analytics Logger auf einem eigenen PC. Dies kann in einer Maschinenanwendung der Fall sein. Alle anderen Tools der Analytics-Toolkette befinden sich auf einem Gerät, einschließlich des MQTT-Message-Brokers.

### **Topologien mit zusätzlichen ASP Clients**

Aktuell sind zwei weitere Clients aus Sicht des Analytics Storage Providers verfügbar. Ein Kommandozeilenbasierter Client, der eine Ausführung aus nahezu jeder Applikation zulässt. Und eine SPS Bibliothek mit der ebenfalls Einfluss auf die Aktionen des Storage Providers genommen werden kann.

## 6 Konfiguration

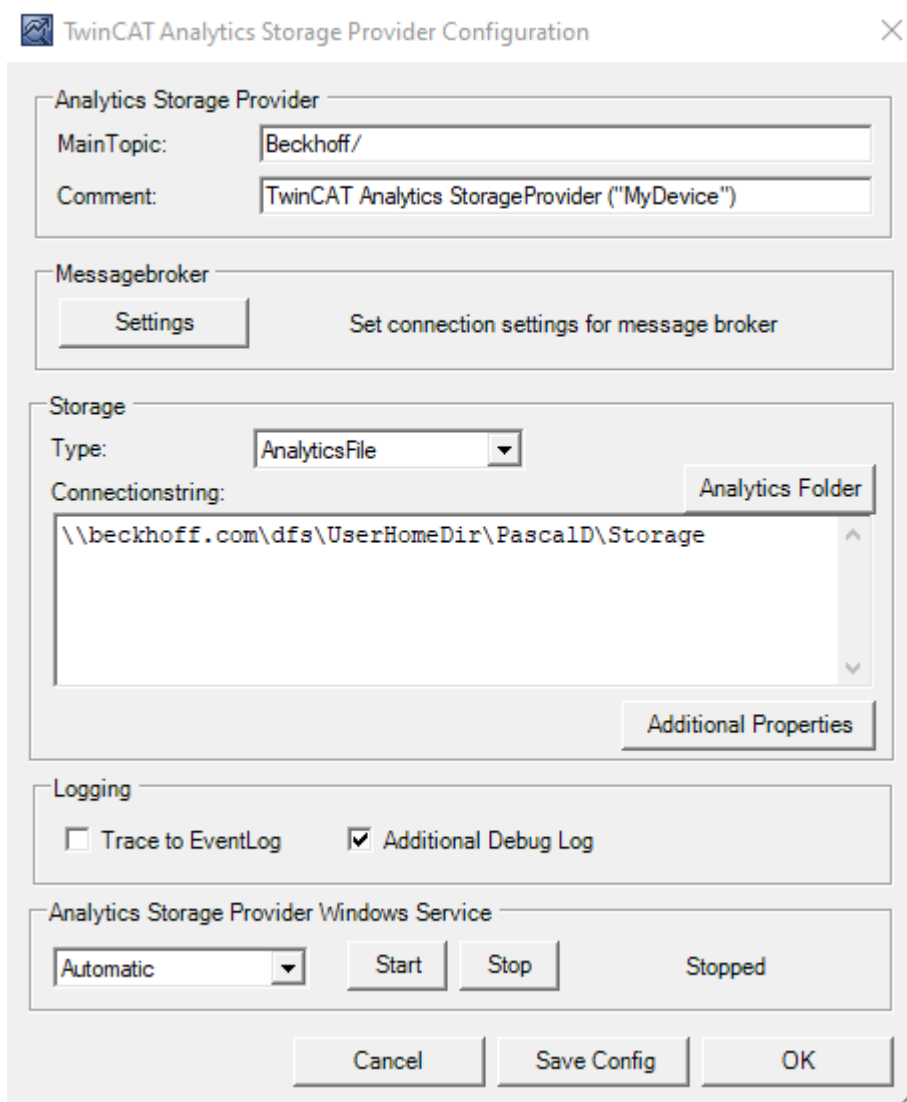
Die Konfiguration des Analytics Storage Providers ist in zwei Hauptteile unterteilt. Der Benutzer muss den Service des Storage Providers und Variablen durch einen Recorder konfigurieren. In diesem Kapitel finden Sie auch die unterstützten Datenbanken und Storages.

### 6.1 Service

Der Windows Service des Analytics Storage Provider benötigt mindestens eine TwinCAT 3 ADS als Grundlage. Das Setup erkennt automatisch, ob ein TwinCAT 3 XAE vorhanden ist oder nicht. Falls vorhanden, installieren Sie auch die SPS-Bibliothek, ansonsten nur den Windows Service.

Nach der Installation finden Sie alles, was Sie benötigen, unter *C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider*. Eine Hierarchieebene tiefer sehen Sie den Ordner WinService, in dem Sie die Datei *TcAnalyticsStorageProvider\_config.exe* finden.

Öffnen Sie diese ausführbare Datei und Sie sehen die Konfigurationsoberfläche.



- Analytics Storage Provider
  - Main Topic:** Legen Sie hier Ihr eigenes Main-Topic fest, um einen Stream mit historischen Daten zu identifizieren.
  - Comment:** Sie können einen Kommentar eingeben, der später im Recorder-Fenster in der Analytics Workbench oder dem Service Tool angezeigt wird.
- Message-Broker
- **Settings:**

Geben Sie hier Ihre Message-Broker-Daten wie URL, Benutzername, Passwort oder Zertifikate an.

Die Einstellungen können überprüft werden. Das Ergebnis wird im folgenden Fenster angezeigt:

Test	Result
✓ Check broker host name [MyMessageBroker.cloudapp.net] IP:172.17.62.145	Success
✓ Check broker ip address [172.17.62.145] (Ping)	Success
✓ Check CA certificate 'MyCA.crt' ExpirationDate:03.12.2036 13:09:38	Success
✓ Connect client to broker	Success

✓ **Connection settings: OK**

- Storage  
**Type:** Wählen Sie hier den Typ Ihrer Datenquelle, wie Analytics File oder Microsoft SQL  
**Connection String:** Sie können Ihren Connection String manuell oder automatisch durch die gegebene Einstellungen-Maske konfigurieren.  
 Für Microsoft SQL siehe:

- Additional Properties  
 In den Additional Properties können zwei weitere Einstellungen vorgenommen werden.

**Max Write Length:**

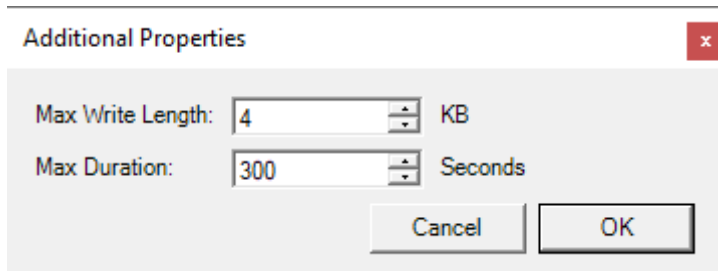
Bei Analytics File: Hier wird die Datenmenge angegeben, welche in einem Aufruf in die .tay Datei geschrieben wird

Bei MsSQL: Hier wird die Datenmenge angegeben, welche in einem tbl\_Data Datensatz gespeichert wird

Max Duration:

Bei Analytics File: Hier wird die Zeit angegeben, nach der eine neue .tay Datei erzeugt wird

Bei MsSQL: Hier ist diese Einstellung nicht relevant



- Analytics Storage Provider Windows Service

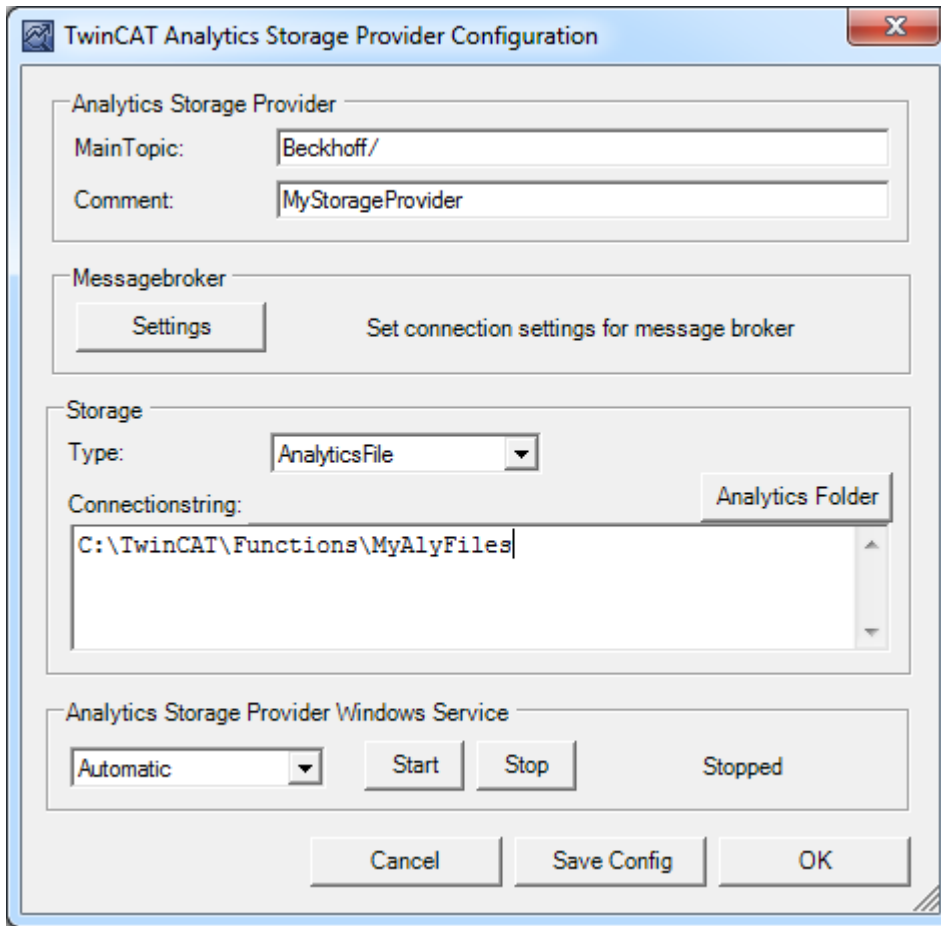
**Status:** Sehen Sie hier den Status und die Standardeinstellungen zum Starten des Windows Service

Selbstverständlich kann die erstellte Konfiguration gespeichert werden. Wenn Sie auf **Save Config** klicken, werden die Einstellungen in TcAlySPWinService.exe.config in *C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider\WinService* gespeichert.

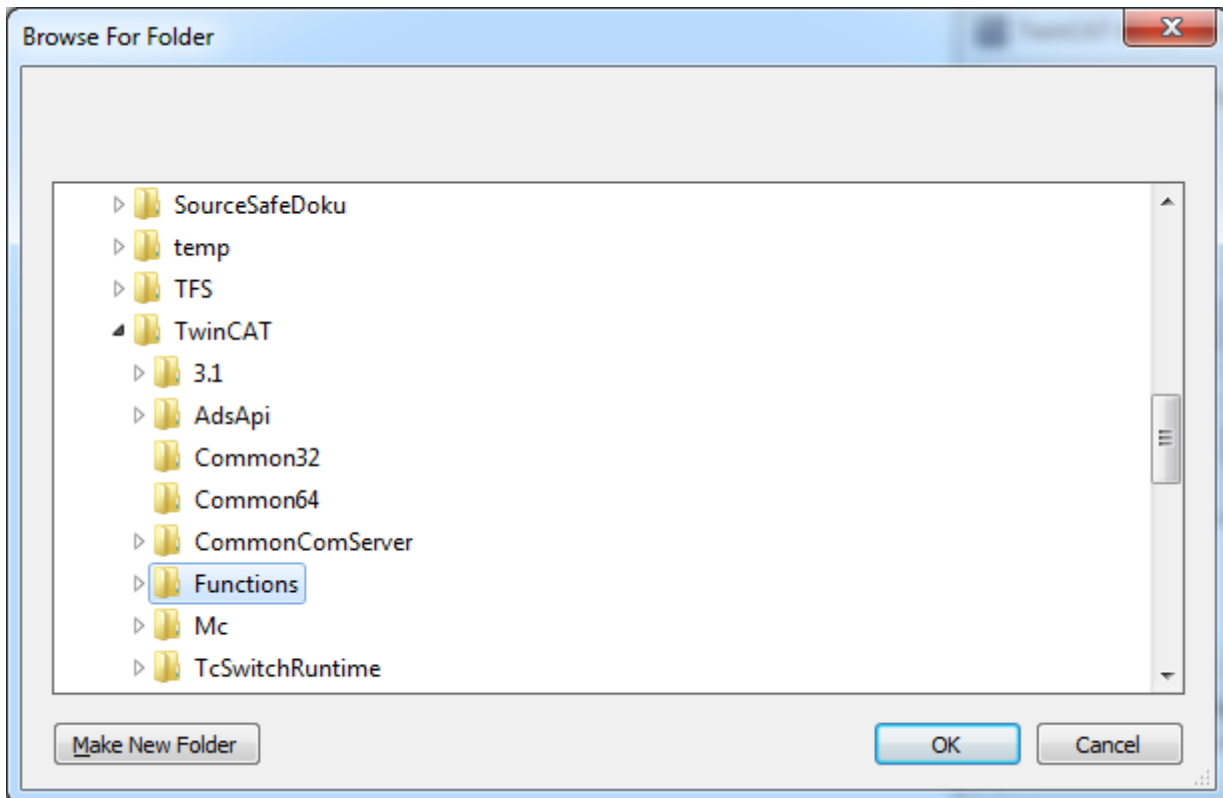
## 6.2 Datenbanken/Storage

### 6.2.1 Analytics Binary File

TwinCAT Analytics Binary File ist ein TwinCAT-spezifischer Storage. Daher ist keine externe Software erforderlich. Sie können diese Speicherart direkt nach der Installation des Analytics Storage Providers nutzen. Es handelt sich um dieselbe Datei, die der TwinCAT Analytics Logger in seiner „Offline“-Konfiguration ohne MQTT-Message-Broker bereitstellt.



Für die Konfiguration müssen Sie lediglich Ihren bevorzugten Ordner auf dem lokalen Gerät, auf dem der Storage Provider läuft, auswählen.

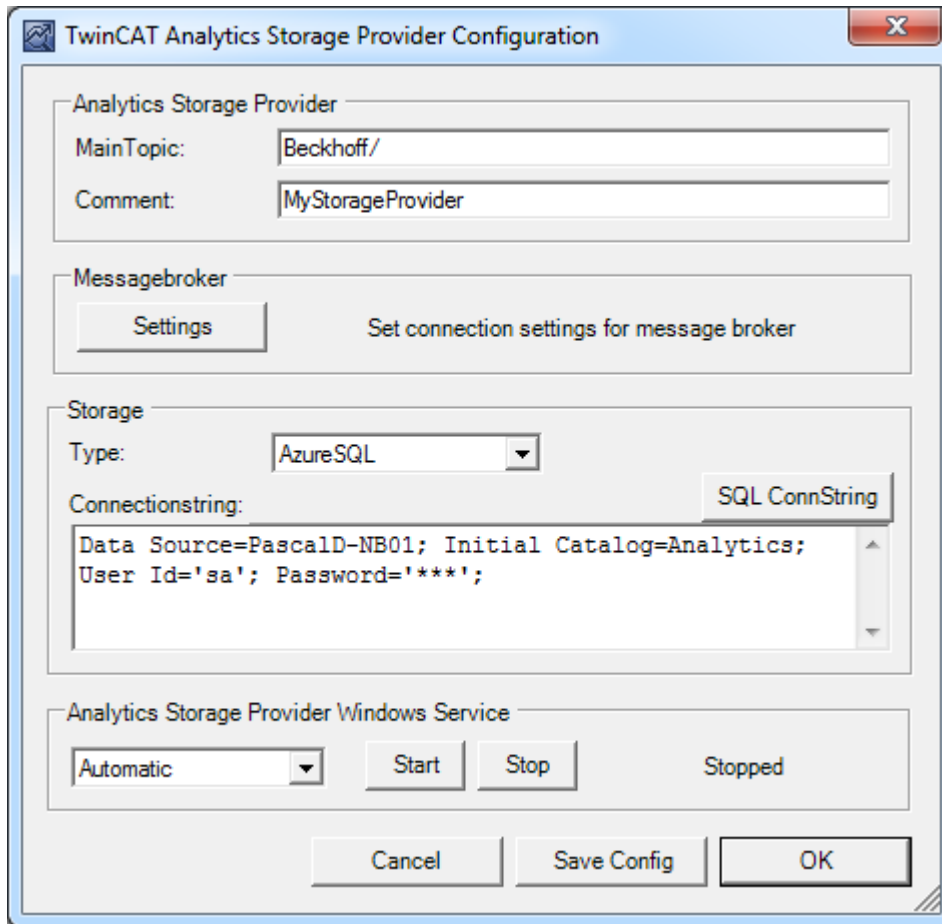


Zur Bestätigung wird der verwendete Ordner im Fenster Connection String angezeigt.

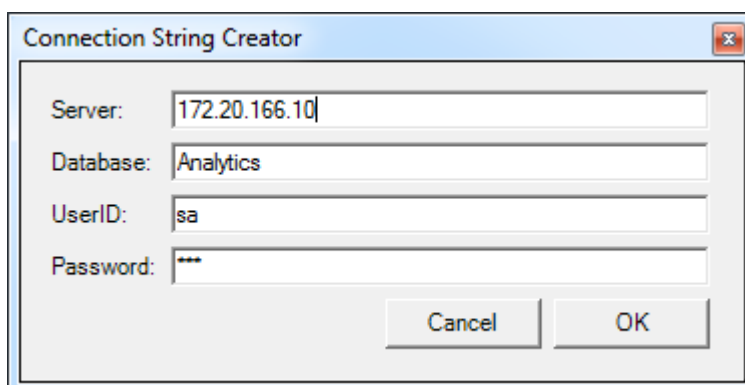


## 6.2.2 Microsoft SQL

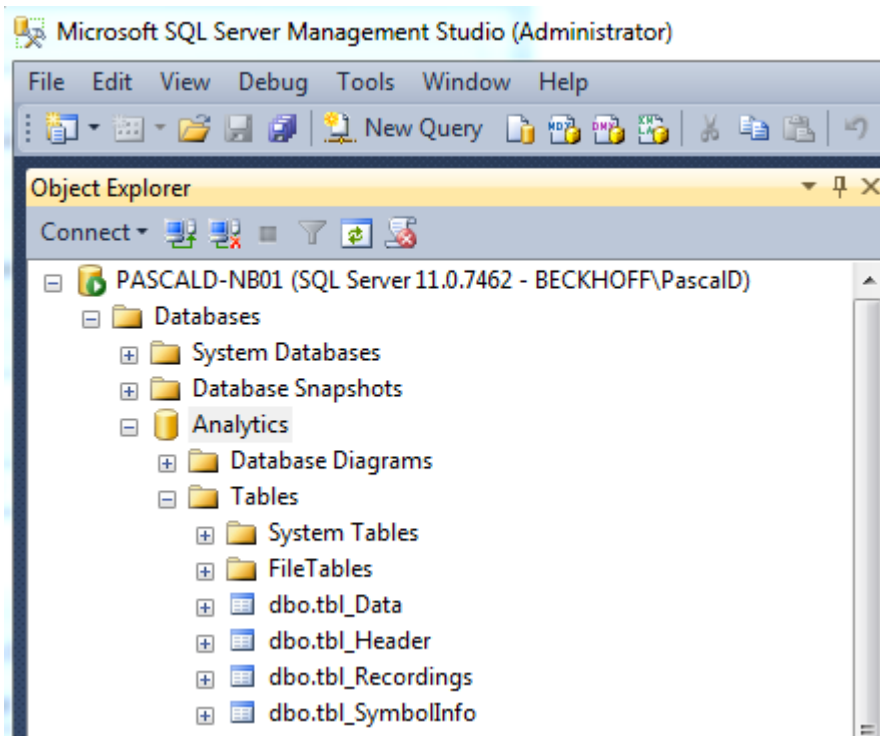
Mit der Unterstützung von Microsoft SQL Server (ebenso Azure SQL) haben Sie eine weitere On-Premises-Lösung für die Speicherung der Analytics-Binärdaten. Sie müssen lediglich den Connection String für Ihren MsSQL Server angeben.



Um dies einfach zu gestalten, können Sie auf die Schaltfläche **SQL ConnString** klicken, um die Eingabemaske zu öffnen. Dort können Sie die Konfigurationseinstellungen vornehmen, auch von Remote-Datenbanken, die durch Netzwerkverbindungen erreichbar sind.

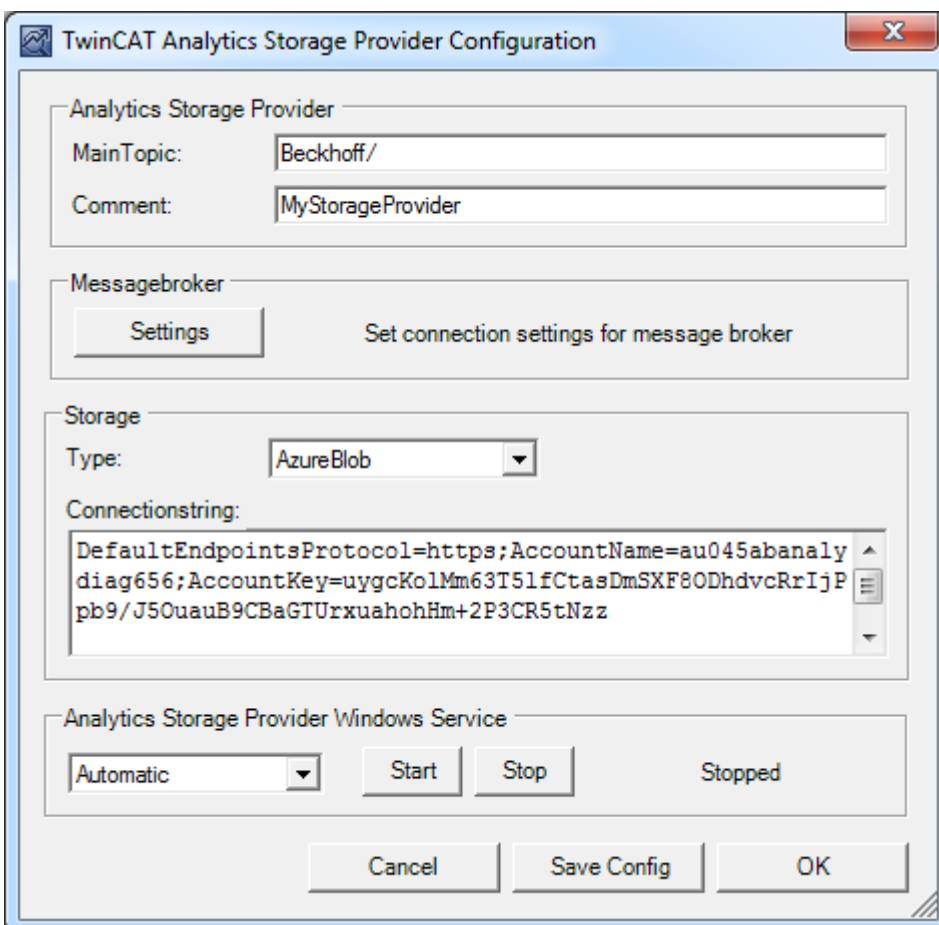


Nach dem Start des WinService und dem Abschluss der allerersten Konfiguration des Recorder-Tools beginnen wir mit der Kommunikation mit der Datenbank. In diesem Moment erstellt der Storage Provider die vier notwendigen Tabellen selbst. Als Beispiel sehen Sie den folgenden Screenshot von Microsoft SQL Server Management Studio.

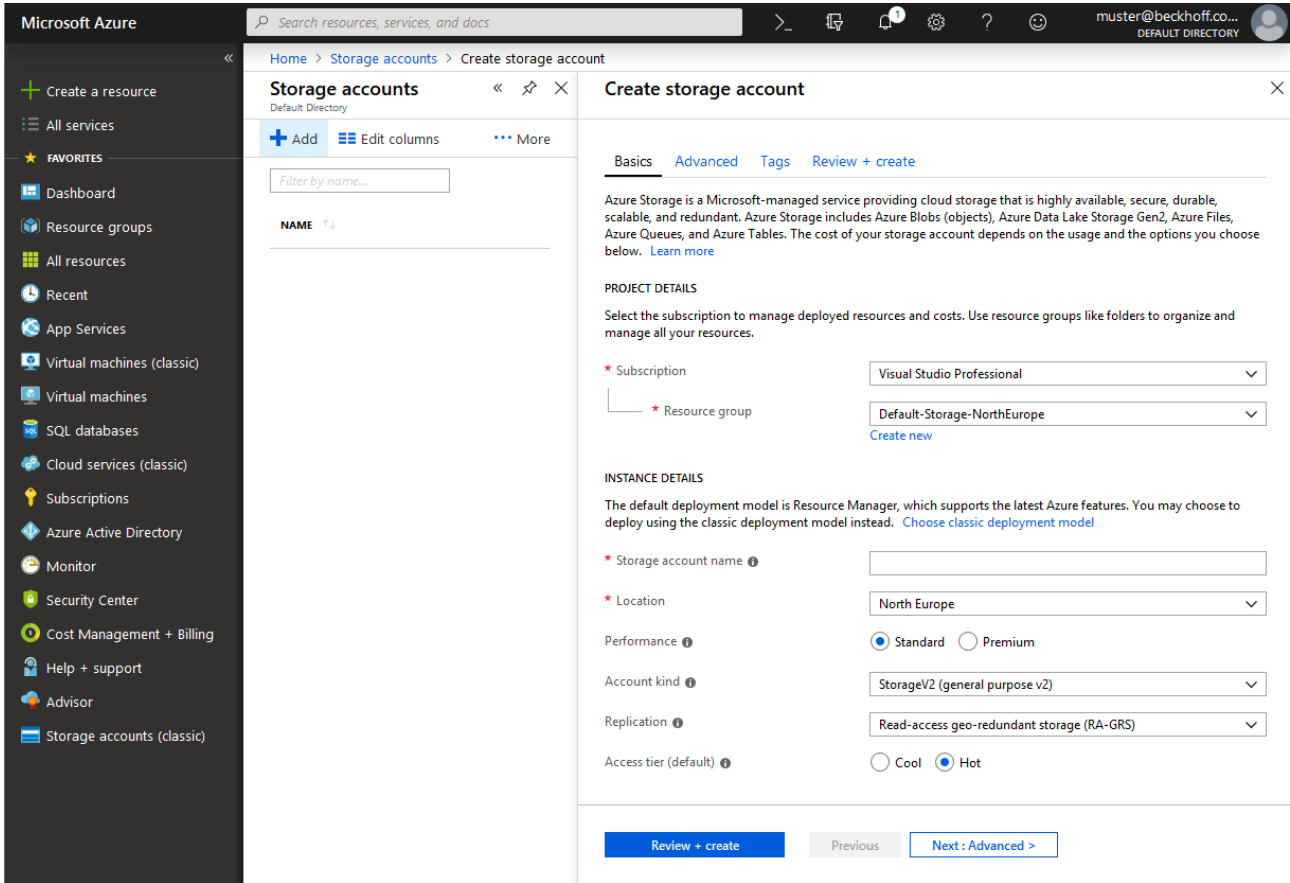


### 6.2.3 Microsoft Azure Blob

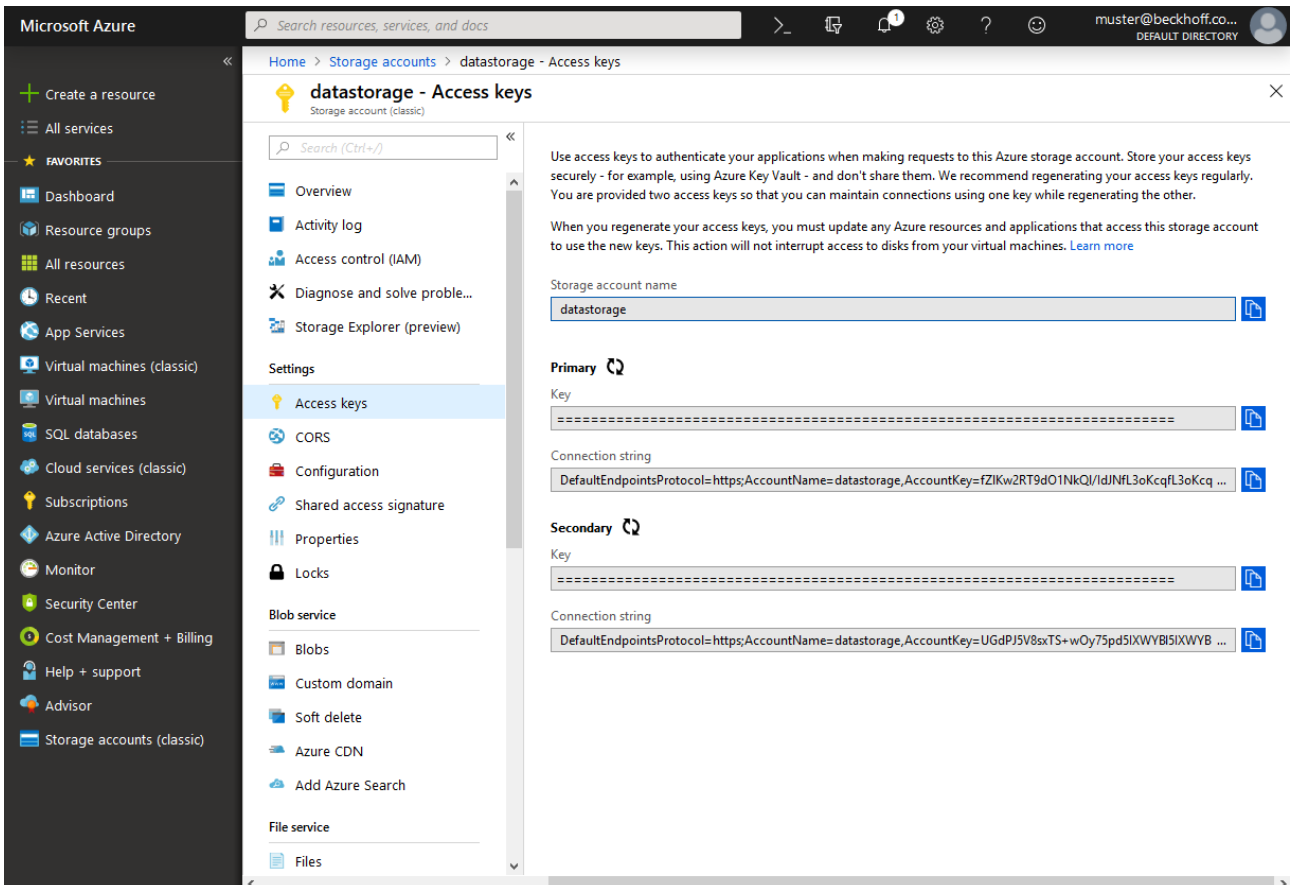
Um den Storage von Microsoft Azure Blob zu nutzen, benötigen Sie ein Microsoft Azure Cloud-Konto. Dort erhalten Sie auch Ihren individuellen Connection String für die Konfiguration des TwinCAT Analytics Storage Providers.



Sie müssen den Connection String in das Beschreibungsfeld kopieren. Der Speicher muss in Azure selbst erzeugt werden. Wählen Sie **Storage accounts (classic)** aus.



Nach der Erzeugung des Storage finden Sie unter **Access keys** den sekundären Connection String. Dieser String muss in der Konfiguration des Analytics Storage Providers verwendet werden.



## 6.3 Recorder

Der Analytics Storage Provider Recorder ist Teil der Analytics Engineering-Setups. Daher finden Sie den Recorder in der Installation der TwinCAT Analytics Workbench und des TwinCAT Analytics Service Tools.

Mit dem Storage Provider Recorder können Aufnahme Definitionen erstellt, gestartet und verwaltet werden. Zusätzlich besteht die Möglichkeit, die Datenspeicher einzelner Analytics Storage Provider zu verwalten. Alle wichtigen Eigenschaften der gefundenen Analytics Storage Providern und historisierten Daten werden übersichtlich dargestellt.

The screenshot shows the 'TwinCAT Analytics Storage Provider Recorder' window. The left pane displays a tree view of recording definitions. The right pane shows details for a selected recording, including storage information and recording times.

Storage	
Topic	TestSignals/Machining Center
TopicAlias	Machining_Center
Layout	9c16bc6f-ef72-f47c-0b8d-2eae14b4328
SystemID	aedf8c92-6306-aa2e-a7fc-64089718415b
SystemAlias	
RecordingAlias	Record_2

Location	
Address	Hülshorstweg 20, 33415 Verl

Record	
RecordID	6
RecordAlias	Record_2
CycleTime	10000 (1,000 ms)
SampleDataSize	89 (0,087 KB)

Record Times	
StartTimestampTicks	133262854558670000
EndTimestampTicks	133262860558700000
StartTimestamp	18.04.2023 11:57:35
EndTimestamp	18.04.2023 12:07:35
Duration	0d 0h 10m 0s 000ms

### Symbolleiste

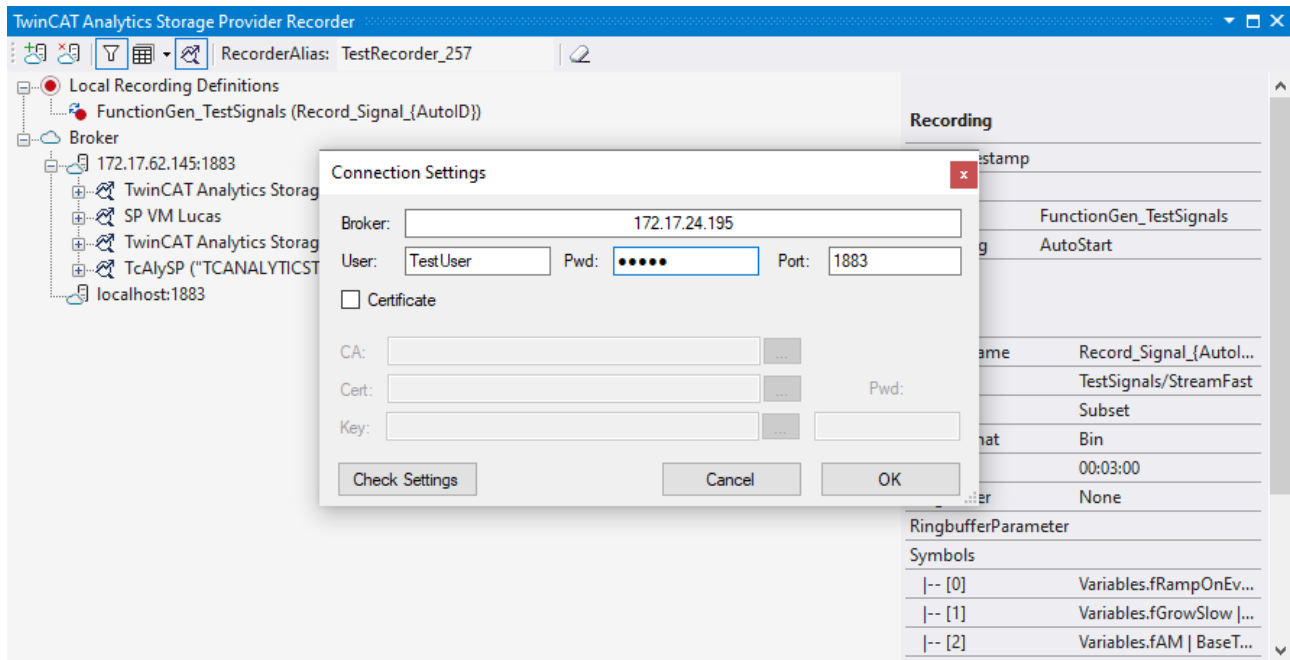
The toolbar contains seven icons: 1. Add new broker, 2. Remove selected broker, 3. Filter view, 4. Select view type, 5. Toggle offline providers, 6. RecorderAlias input, 7. Remove error messages.

1	Neuen Broker hinzufügen
2	Ausgewählten Broker entfernen
3	Anzeige Filter (Alle / Meine aktiven Aufnahmen anzeigen)
4	Anzeige Art wählen (Alias / Topics)
5	Offline Analytics Storage Provider anzeigen / ausblenden
6	„RecorderAlias“ – Gruppierungsname von Aufnahmen
7	Meldungen aus Fehlerliste entfernen

### Recorder Fenster einrichten

Vergeben Sie zunächst einen „RecorderAlias“. Dieser hilft die gestarteten Aufnahmen zu gruppieren und seine selbst gestarteten wieder zu finden. Mit dem Filter können auch fremd gestartete Aufnahmen angezeigt werden.

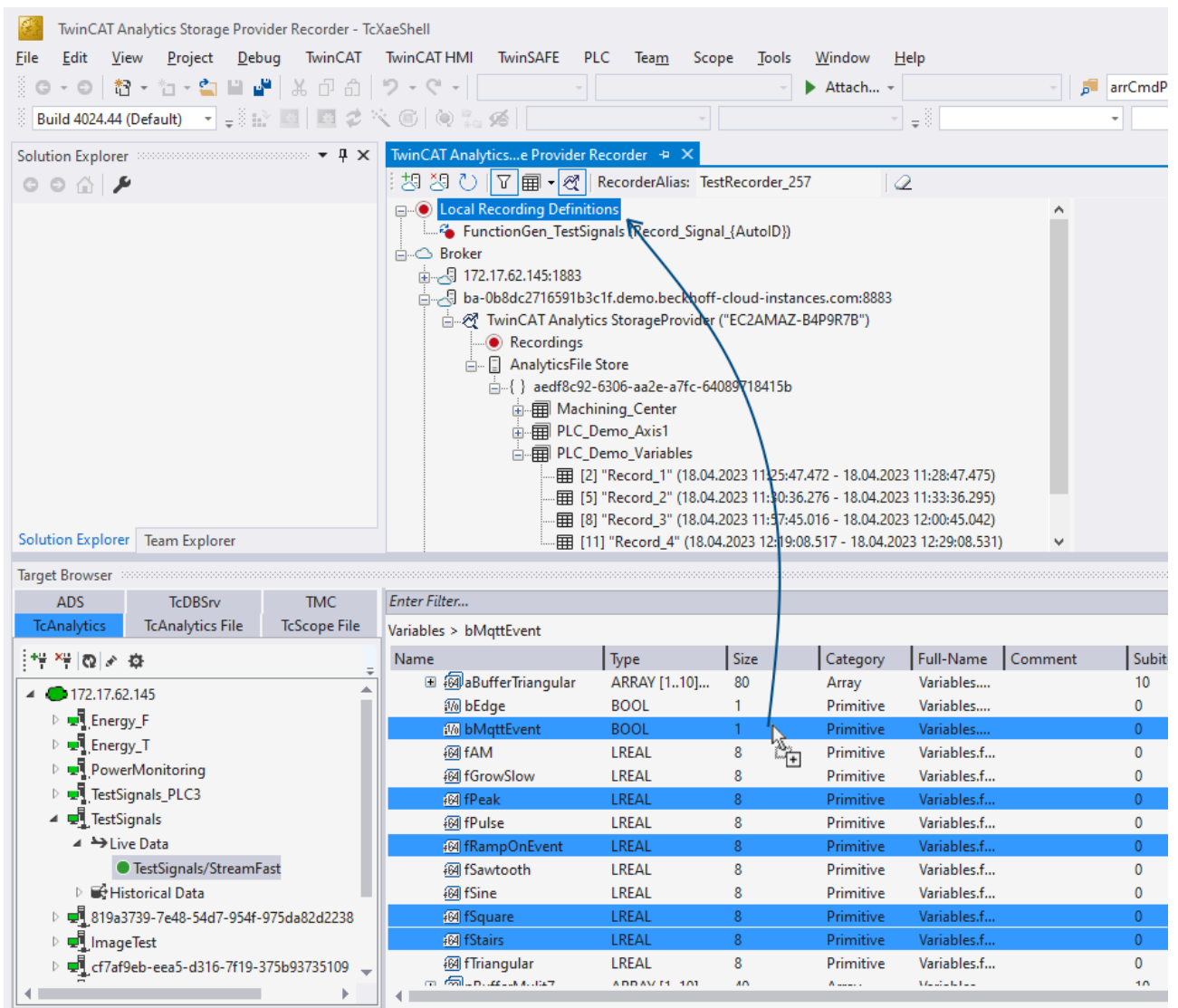
Danach können ein oder mehrere Broker eingerichtet werden. Dies geschieht über die schon bekannte Eingabemaske für MQTT-Verbindungseigenschaften.



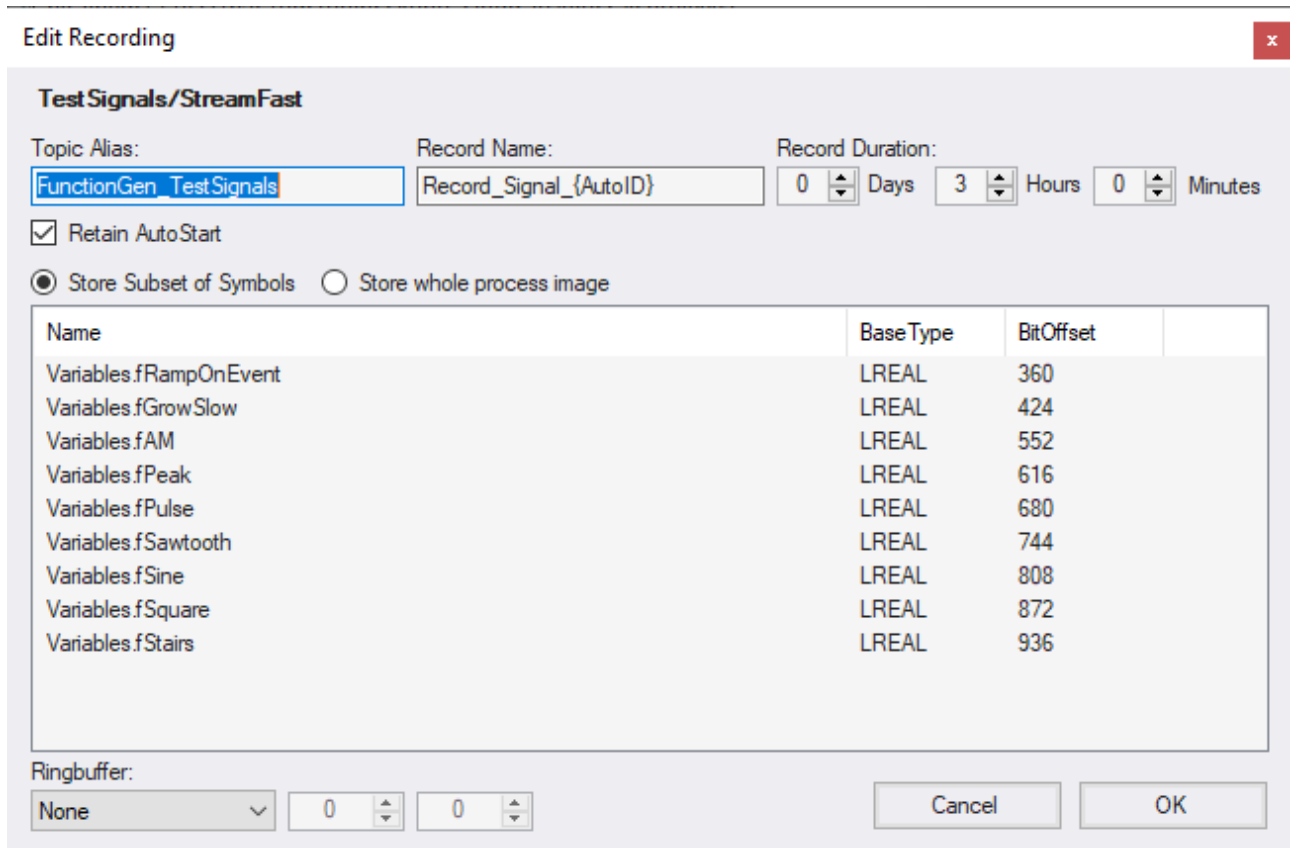
Sobald eine Verbindung zum Broker hergestellt werden konnte, werden alle darauf verbunden Analytics Storage Provider aufgelistet.

### Aufnahmedefinition erzeugen

Zur Konfiguration der Aufzeichnung wählen Sie Ihr Ziel im Target Browser aus. Klicken Sie auf **Live Data** und wählen Sie eine oder mehrere Variablen durch Mehrfachauswahl aus und ziehen Sie sie per Drag-and-Drop auf den **Local Recording Definitions** Knoten im Recorder Fenster.



Im Recorder können Sie die ausgewählten Variablen oder das vollständige Quell-Prozessabbild der Variablen hinzufügen.

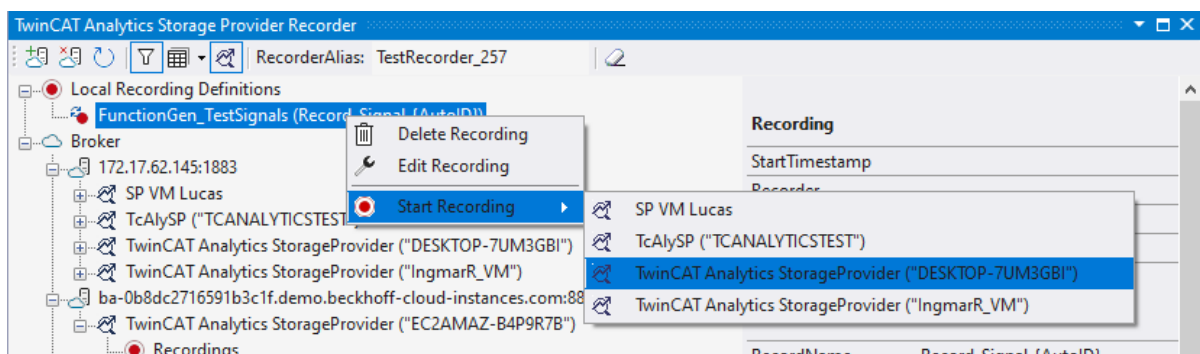


Sie können auch Aufzeichnungsnamen und eine Dauer konfigurieren (anderenfalls läuft die Aufzeichnung endlos, bis sie manuell gestoppt wird). Ein Ringpuffer kann nach Storage oder Zeit eingestellt werden.

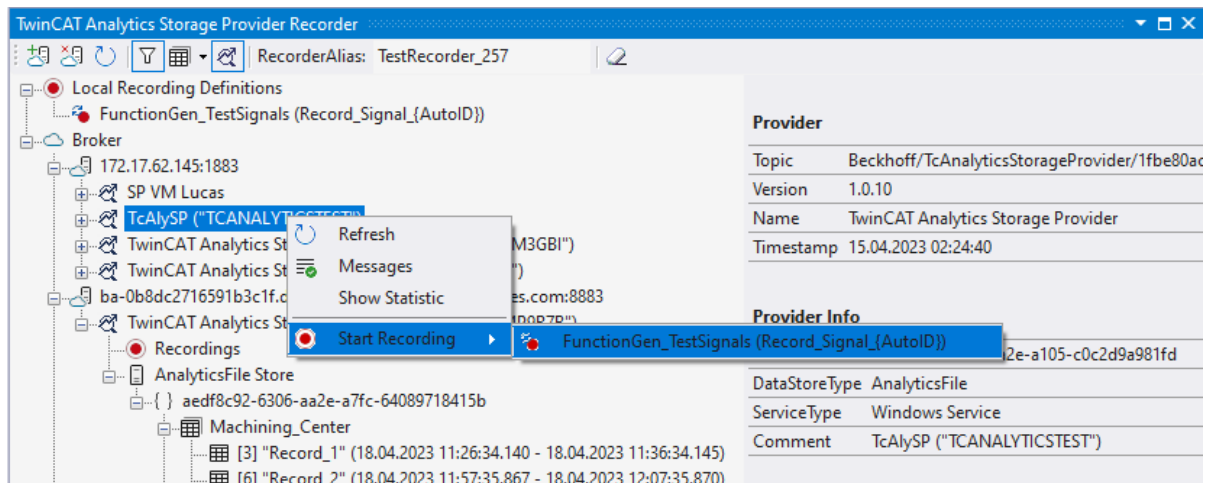
**Aufnahme starten**

Es gibt drei verschiedene Wege, um eine Aufnahme zu starten.

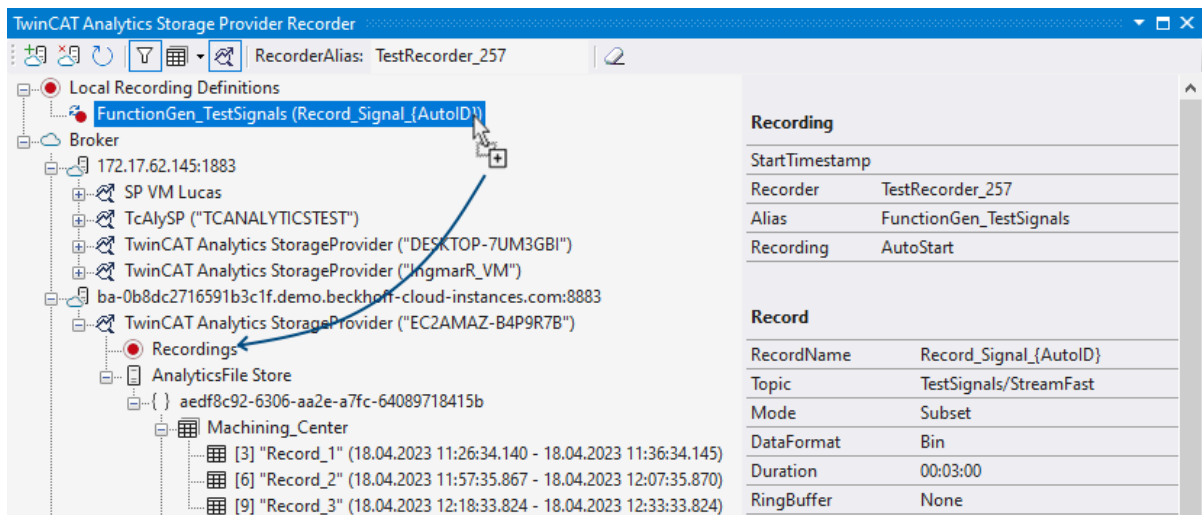
1. Über das Kontextmenü an einem lokalem Aufnahmedefinition Knoten. Hier kann der gewünschte Analytics Storage Provider ausgewählt werden, welcher die Daten aufzeichnen soll. Es werden nur die Storage Provider angezeigt, die auch Zugriff auf das ausgewählte Topic haben.



- Über das Kontextmenü am Analytics Storage Provider Knoten. Hier kann die gewünschte Aufnahmedefinition ausgewählt werden. Es werden nur die Definitionen angezeigt, die auch vom Analytics Storage Provider verarbeitet werden können.



- Per Drag-and-Drop des lokalen Aufnahmedefinition Knoten. Die gewünschte Aufnahmedefinition kann auf den gewünschten Analytics Storage Provider per Drag-and-Drop gestartet werden. Sollte der Storage Provider die Aufnahmedefinition nicht ausführen können, wird eine Fehlermeldung in der Fehlerliste ausgegeben.



## 6.4 Arbeit mit historischen Daten

Historische Daten können mit der Analytics Workbench oder dem Analytics Service Tool analysiert werden. Um Ihre aufgezeichneten Daten anzuzeigen, benötigen Sie den TwinCAT Target Browser.

### Auswahl der Daten aus dem TwinCAT Target Browser

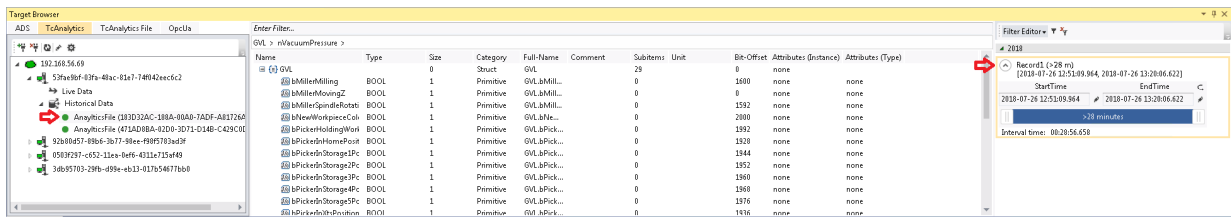
Die historischen Daten können direkt aus dem Target Browser zu einem Eingang eines Analysealgorithmus gezogen werden.

- Zunächst müssen Sie in der linken Ecke des Target Browsers auf **TcAnalytics** klicken. Dort können Sie Ihren konfigurierten Broker sehen, der die Live- und historischen Daten Ihrer verschiedenen Geräte auflistet. Dies sollte wie in der folgenden Abbildung aussehen.

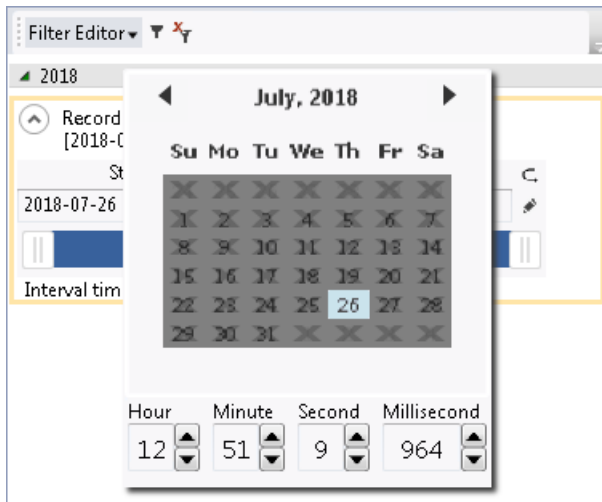




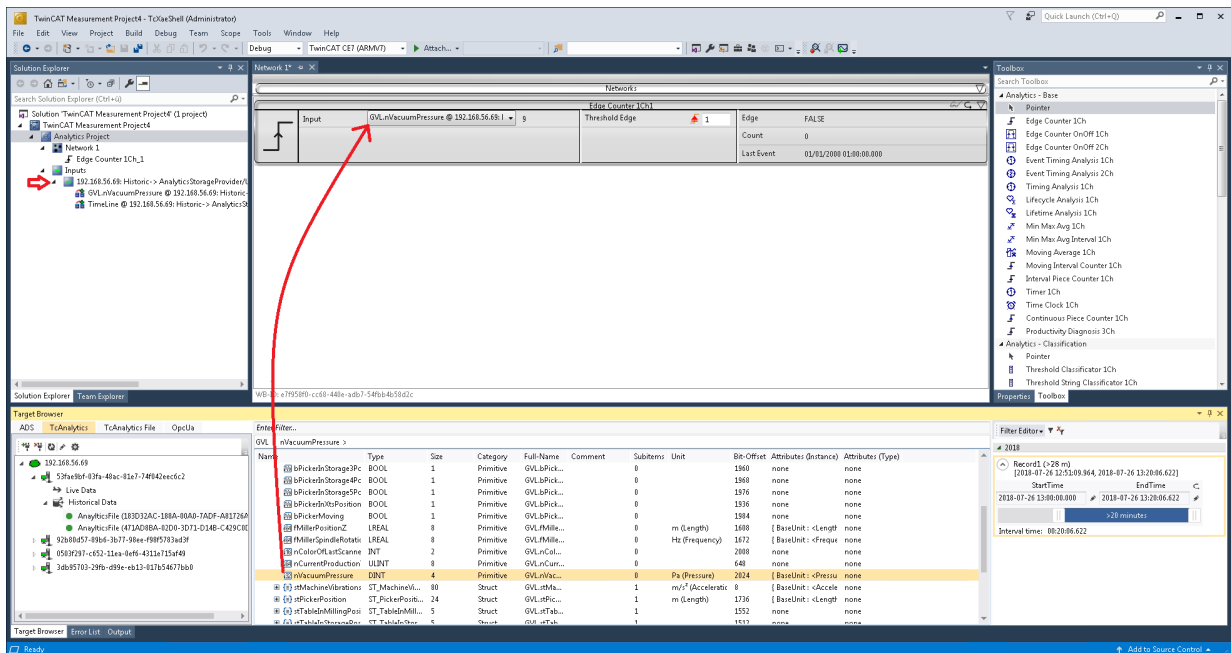
- Gehen Sie zu dem historischen Stream, den Sie erstellt haben, und wählen Sie die Aufzeichnung aus, die analysiert werden soll. Alle Ihre Aufzeichnungen sind rechts im Fenster **Record** aufgelistet. Standardmäßig wird die letzte Aufzeichnung ausgewählt.



- Wenn Sie live aufzeichnen, wird der Zeitbereich der Aufzeichnung alle paar Sekunden aktualisiert. Standardmäßig wird der gesamte Zeitbereich einer Aufzeichnung verwendet. Außerdem können Sie die Start- und Endzeit bearbeiten, um Ihren gewünschten Datenbereich zu analysieren. Dies kann mit einem Schieberegler, Textfeldern oder in einer grafischen Kalenderansicht erfolgen. Wenn Sie auf das Symbol rechts neben den Textfeldern klicken, wird die Kalenderansicht angezeigt.



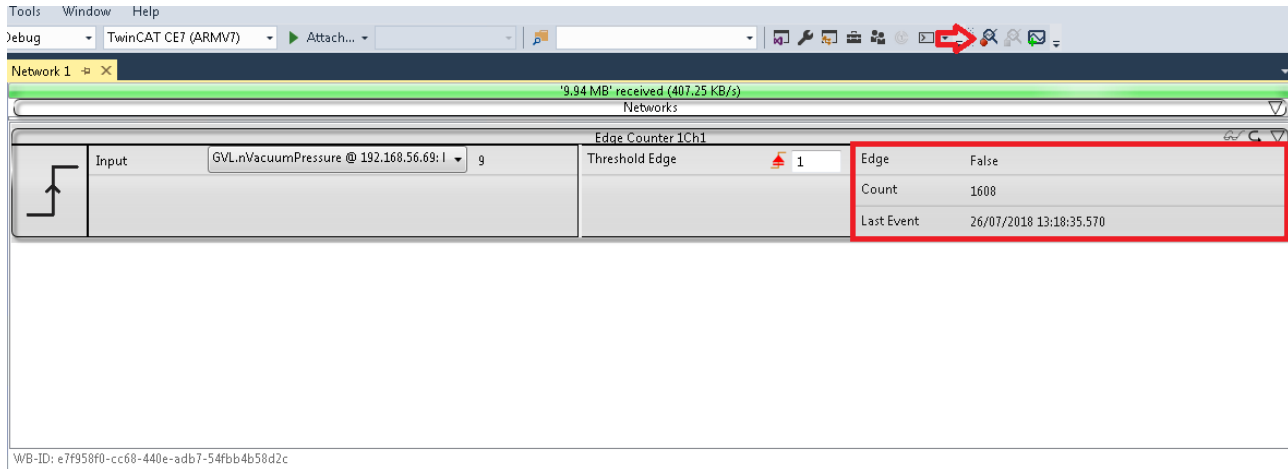
- Nach diesen Schritten können Sie ein Symbol ebenso wie bei den Symbolen der Live-Daten per Drag-and-Drop zu einem Eingang eines Algorithmus ziehen.



- ⇒ Anschließend wird eine neue Eingangsquelle für Ihren historischen Stream generiert und kann im Projektmappen-Explorer Ihres Visual Studios® angezeigt werden. Als Erstes werden das gezogene Symbol und ein Zeitstempel der aktuellen Gerätezeit unter diesem Stream aufgelistet. Auch neue gezogene Symbole dieses Streams werden dort aufgelistet.

## Analyse Ihrer historischen Daten im Analytics-Konfigurator

Um Ihre historischen Daten zu analysieren, drücken Sie auf die Schaltfläche **Start Analytics**. Anders als bei der Analyse von Live-Daten erscheint ein grüner Fortschrittsbalken. Die Geschwindigkeit Ihrer Analyse hängt von der Länge Ihrer Aufzeichnung, der Anzahl und Größe Ihrer Symbole sowie von Ihrer Breitbandgeschwindigkeit zum Broker ab. Die Analyse stoppt automatisch, wenn der Fortschrittsbalken am Ende angelangt ist. Die Ergebnisse bleiben sichtbar.



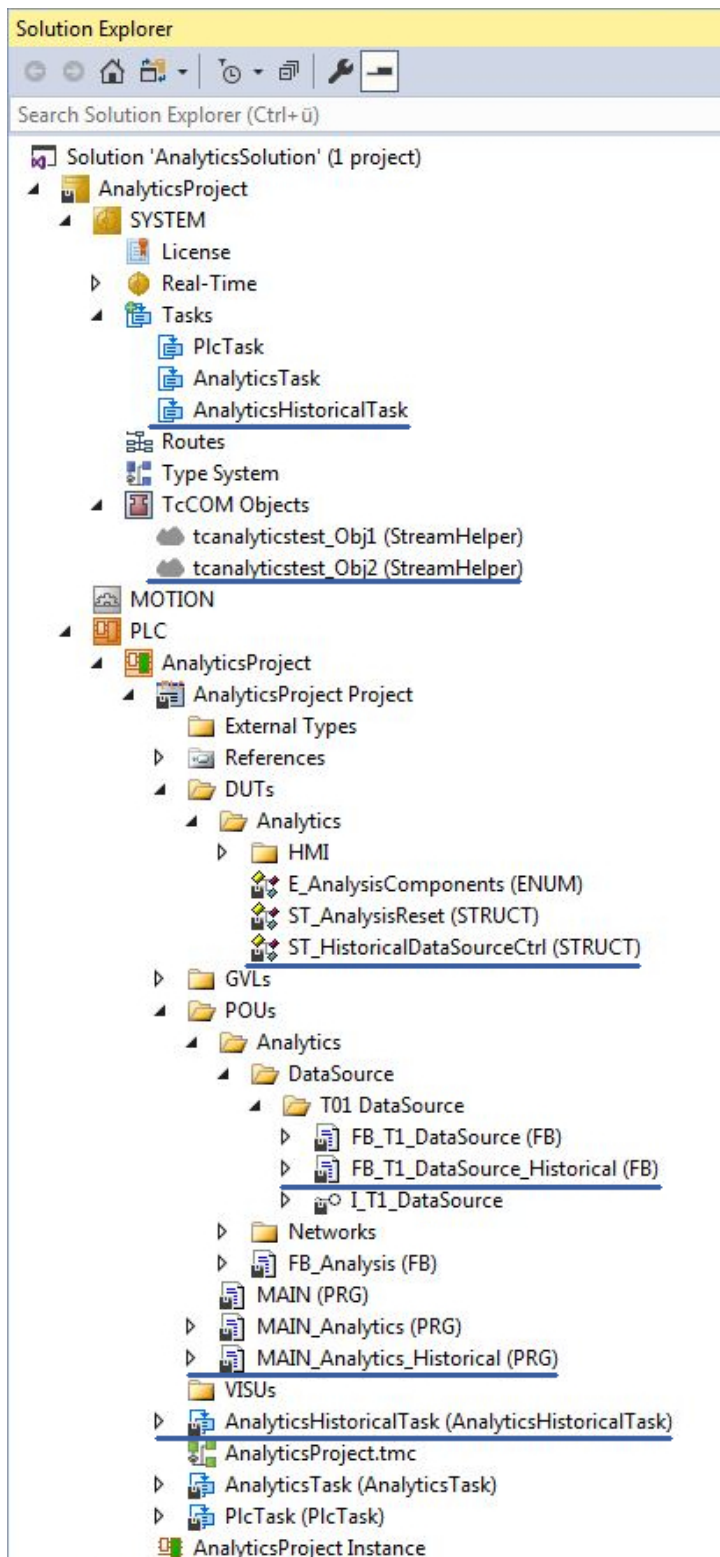
## Analyse Ihrer historischen Daten in Ihrer Analytics Runtime

Sie können die Konfiguration mit Ihren historischen Daten einer Analytics Runtime (SPS) bereitstellen. Zusätzlich zu den historischen Daten werden auch die Live-Daten analysiert. So können Sie zwischen ihnen wechseln und verlieren keine Live-Daten durch das Streamen historischer Daten. Der Grund hierfür ist, dass sie in zwei verschiedene Tasks getrennt sind. Der Start der Analyse historischer Daten muss ausgelöst werden.

### ● Rechenzeit bei historischen Daten

**i** Anders als beim Analytics-Konfigurator dauert die Analyse historischer Daten in der SPS ähnlich lange wie die ursprüngliche Aufzeichnung der Daten. In Abhängigkeit der Paketgröße und der eingestellten Abtastrate kann die Abarbeitung der Daten gegenüber der Aufzeichnung verkürzt werden. Zyklusüberschreitungen aufgrund zu großer Pakete müssen jedoch beachtet werden.

Hauptunterschiede der Ordnerstruktur im erstellten SPS-Projekt:



### HINWEIS

#### Realisierung der Logik in Ihrer TwinCAT HMI

Die Vorbereitung und das Schreiben von Werten in Ihrer SPS dienen Testzwecken. Es wird empfohlen, diese und andere Logik im SPS-Code bei Bedarf mit Interaktionen von Ihrer TwinCAT HMI-Anwendung zu realisieren.

Sie können die Analyse historischer Daten durch Auslösen von **bGetHistoricalData** in **stCtrl\_T1** starten.

Der Abbruch erfolgt durch Auslösen von **bCancelHistoricalData**.

Dies kann in der Datei **MAIN\_Analytics\_Historical** wie in der folgenden Abbildung erfolgen:

Expression	Type	Value	Prepared value	Address	Comment
stCtrl_T1	ST_HistoricalDa...				
bGetHistoricalData	BOOL	FALSE	TRUE		
bCancelHistoricalData	BOOL	FALSE			
bError1	BOOL	FALSE			

Um für Ihr HMI Dashboard zwischen den Ergebnissen der Live-Daten und historischen Daten zu wechseln, können Sie das Symbol **bHistorical** in der GVL **AnalyticsHMI** setzen. Mit dieser Möglichkeit benötigen Sie keine zusätzlichen Controls, um historische Daten anzuzeigen (natürlich sind auch eigene Controls für historische Daten möglich). Die Analyse der Live-Daten wird nicht durch das Aufrufen einiger historischer Daten unterbrochen. Nach der Anzeige der historischen Daten können Sie zu den aktuellen Live-Ergebnissen zurückwechseln. Dieser Wechsel betrifft nur die Variablen in Ihrer GVL.

Expression	Type	Value	Prepared value	Address	Comment
bHistorical	BOOL	FALSE	TRUE		Switch
bN1_Network1_Reset	BOOL	FALSE			Reset
bN1_M1_EdgeCounter1Ch_1_Reset	BOOL	FALSE			
stHMI_N1_Network1	ST_HMI_N1_Network1				HMI S

## 6.5 Konsolen- Konfigurator/Client

Der Analytics Storage Provider kann neben dem grafischen Konfigurator und dem Recorder auch über eine Konsole bedient werden. Dadurch kann die Konfiguration und der Zugriff auf einen Storage Provider neben Windows auch unter TwinCAT/BSD durchgeführt werden. Außerdem können mithilfe der Konsolen-Applikation Batch Files zur Steuerung [▶ 65] des Analytics Storage Providers erzeugt werden.

Nach dem Start des Konsolen-Clients stehen vier Optionen zur Auswahl:

```
#####
##### TWINCAT ANALYTICS STORAGE PROVIDER #####
#####

----- Main Menu -----
-----

1 : Local Provider Configurator
2 : Provider Client
c : Clear Screen
q : Exit Program

Please enter a task:~ $
```

1	Öffnet den Konsolen-Konfigurator [▶ 61] für den lokalen Analytics Storage Provider
2	Öffnet den Analytics Storage Provider Client [▶ 62]
c	Leert die Historie des Konsolen-Konfigurators/Clients
q	Schließt den Konsolen-Konfigurator/Client

Durch Eingabe eines der Kennzeichen und bestätigen mit der **[Enter]**-Taste wird die entsprechende Funktion ausgeführt.

### 6.5.1 Konfigurator

In diesem Menü kann der lokale Analytics Storage Provider konfiguriert werden. Dazu stehen folgende, zusätzliche Eingaben zur Verfügung:

<b>1</b>	Startet einen Dialog zum Konfigurieren des lokalen Analytics Storage Providers
<b>2</b>	Gibt die Konfiguration des lokalen Analytics Storage Providers aus
<b>10</b>	Startet den Analytics Storage Provider mit der lokalen Konfiguration
<b>11</b>	Stoppt den lokalen Analytics Storage Provider
<b>12</b>	Gibt den Status des Analytics Storage Providers aus
<b>b</b>	Wechselt zurück zum Hauptmenü

Die Konfigurationsparameter sind dieselben wie in dem grafischen Konfigurator des Analytics Storage Providers. Zur Erstellung der Konfiguration werden die Konfigurationsparameter nach der Eingabe „1“ einzeln entgegengenommen:

```
##### TWINCAT ANALYTICS STORAGE PROVIDER #####
##### TWINCAT ANALYTICS STORAGE PROVIDER #####

----- Main Menu Local Provider Configurator -----
-----
1 : Create Local Provider Config
2 : Read Local Provider Config

10 : Start Local Provider Service
11 : Stop Local Provider Service
12 : Status Local Provider Service

b : Back to Start
c : Clear Screen
q : Exit Program

Please enter a task:~ $ 1

Analytics Storage Provider Settings:
Please enter Provider Main Topic:~ $ Beckhoff/
Please enter Provider Comment:~ $ TwinCAT Analytics StorageProvider (Test)
Trace to EventLog (y/n):~ $ y

Additional Debug Log (y/n):~ $ y

Messagebroker Settings:
Please enter Broker hostname/IP:~ $ localhost
Please enter Broker Port:~ $ 1883
Please enter Username:~ $ TestUser
Please enter Password:~ $ *****
Use certificates for connect? (y/n):~ $ n

Storage Settings:
(1) - ANALYTICSFILE
(2) - MSSQL / AZURESQL
(3) - AZUREBLOB
Please choose id from storage type:~ $ 1

Please enter Storage path:~ $ C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider\Storage
Please enter Max Write Length in bytes:~ $ 2048
Please enter Max Duration in seconds:~ $ 120
Configuration successfully saved!

----- Main Menu Local Provider Configurator -----
-----
1 : Create Local Provider Config
2 : Read Local Provider Config

10 : Start Local Provider Service
11 : Stop Local Provider Service
12 : Status Local Provider Service

b : Back to Start
c : Clear Screen
q : Exit Program

Please enter a task:~ $
```

Bei der Auswahl des Speichertyps ist **AnalyticsFile** der Standard Speichertyp bei einer Falscheingabe. Durch die **[ESC]**-Taste kann der Dialog jederzeit abgebrochen und zurück in das Konfigurations-Menü gewechselt werden.

## 6.5.2 Client

In dem Analytics Storage Provider Client ist es möglich, eine Verbindung zu einem Analytics Storage Provider herzustellen. Dieser muss dabei nicht zwingend lokal auf dem Gerät laufen, sondern kann auch über einen externen MQTT-Broker angesprochen werden. Für die Verwendung des Clients ist es also nicht

zwingend notwendig einen Analytics Storage Provider lokal zu konfigurieren. Auf einem verbundenen Analytics Storage Provider können anschließend Aufnahmen, sowie historische Datenstreams an ein konfigurierbares MQTT-Topic gestartet und gestoppt werden.

Dazu stehen folgende Eingaben zur Verfügung:

1	Stellt eine Verbindung mit dem MQTT-Broker aus der lokalen Konfiguration her und selektiert den konfigurierten Analytics Storage Provider
2	Startet einen Dialog, um eine Verbindung mit einem MQTT-Broker herzustellen
3	Schließt die Verbindung zu dem aktuellen MQTT-Broker
10	Stellt die Analytics Storage Provider zur Auswahl bereit, die unter dem MQTT-Broker verfügbar sind
20	Liest eine Konfigurations-Datei von einem <u>Analytics Storage Provider Recorder</u> , mit den darin konfigurierten Recordings ein
31	Startet eine Aufnahme
32	Stoppt eine Aufnahme anhand des Alias und dem MQTT-Topic
33	Startet einen historischen Datenstream
34	Stoppt einen historischen Datenstream anhand des Ergebnis-MQTT-Topics
35	Aktualisiert Streaming-Parameter eines laufenden historischen Datenstreams
36	Prüft, ob eine Aufnahme aktiv ist
40	Beendet alle aktiven Aufnahmen
41	Beendet alle historischen Datenstreams

Zur Verwendung des Analytics Storage Provider Clients muss zunächst eine Verbindung zu einem MQTT-Broker aufgebaut werden. Durch die Eingabe „2“ wird ein Dialog gestartet, in welchem die bereits konfigurierten MQTT-Broker zur Auswahl gestellt werden. Dort kann durch die Eingabe „0“ ein neuer MQTT-Broker konfiguriert und verbunden werden. Anschließend kann durch die Eingabe „10“ im Client-Hauptmenü ein Analytics Storage Provider ausgewählt werden, welcher unter dem MQTT-Broker verfügbar ist. Alternativ kann durch die Eingabe „1“ im Client-Hauptmenü direkt eine Verbindung mit dem MQTT-Broker und Analytics Storage Provider aus der lokalen Konfigurationsdatei hergestellt werden.

Nach dem erfolgreichen Verbindungsaufbau, wird in der Prompter-Anzeige eine Information über den verbundenen MQTT-Broker und den selektierten Analytics Storage Provider vor der Eingabeaufforderung angezeigt:

```
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter a task:~ $
```

(MQTT-Broker : Analytics Storage Provider)

Zum Starten von Aufnahmen wird mit der Eingabe „31“ ein Dialog gestartet. Dabei besteht die Möglichkeit, eine bereits erstellte Aufnahme zu starten, insofern schon Aufnahmen konfiguriert oder über eine Recorder-Konfigurationsdatei eingelesen wurden. Neben den aufgeführten Aufnahmen kann durch die Eingabe „0“ auch eine neue Aufnahme konfiguriert und gestartet werden:

```

RecorderAlias:Console ASP Client ("MARCT-NB01") - [ffca662c-0000-0000-0000-80d643e8d241] ← 1
----- Main Menu Provider Client -----
1 : Use Local Provider
2 : Connect
3 : Disconnect

10 : Select Provider

20 : Load Config (RecorderSettings)

31 : Start Recording
32 : Stop Recording
33 : Start Historical Stream
34 : Stop Historical Stream
35 : Set Historical Stream Parameter
36 : Is Recording Active

40 : Cancel All Recordings
41 : Cancel All Historical Streams

b : Back to Start
c : Clear Screen
q : Exit Program

[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter a task:~ $ 31
Recordings:
(0) - [Create new recording]
(1) - "TcBSD_ASP_Recording" (ASP_Record)
(2) - "TcBSD_ASP_Alias" (TcBSD_Record)
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please choose id from record config:~ $ 0

[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter topic:~ $ TestSignals/StreamFast
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter recording alias:~ $ ASP_Recording
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter recordname:~ $ AnalyticsSP_Record
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter duration in minutes:~ $ 120
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter ringbuffer [None|TimeBased|DataBased]:~ $ None
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter dataFormat [Bin|Json]:~ $ Bin
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter recording mode [All|Subset]:~ $ All
Recording command "ASP_Recording" send to Provider.
Recording "ASP_Recording" is running.

```

Die Konfigurationsparameter entsprechen dabei den bekannten Parametern aus dem grafischen Analytics Storage Provider Recorder. Die Default-Werte können bei Bedarf gelöscht und durch individuelle Eingaben ersetzt werden. Bei dem **recording mode** kann nach der Eingabe „Subset“ eine Untermenge der Symbolik von den Aufnahmedaten definiert werden. Direkt nach der Konfiguration wird ein Kommando zum Start der Aufnahme an den verbundenen Analytics Storage Provider gesendet. Laufende Aufnahmen können über die Eingabe „32“ gestoppt werden. Dabei werden sie durch das MQTT-Topic, von dem die Daten kommen und durch den Aufnahme Alias referenziert. Wenn eine Aufnahme von einem anderen Client gestoppt werden soll, muss zusätzlich die entsprechende **Recorder-Guid** angegeben werden. Die **Recorder Guid** wird gemeinsam mit dem **Recorder-Alias** über den Eingabemöglichkeiten im Client-Hauptmenü angezeigt (rote 1).

Aufnahme-Konfigurationen, die in dem Konsolen Client erstellt wurden, werden nicht persistiert. Nach dem Schließen des Clients ist die Liste der Aufnahme-Konfigurationen also nicht mehr vorhanden. Daher kann das Einlesen von Recorder-Konfigurationsdateien (Eingabe „20“ im Client-Hauptmenü) sehr hilfreich sein. Die Konfigurationsdatei eines Recorders wird unter Windows-Systemen unter dem Pfad `C:\Users\***\AppData\Roaming\Beckhoff\TwinCAT Analytics Storage Provider` abgelegt (\*\*\* durch den entsprechenden User ersetzen).

Die historisierten Daten des Analytics Storage Providers können über die Eingabe „33“ als Datenstream an ein definierbares Ergebnis-MQTT-Topic übermittelt werden. Dabei wird ebenfalls ein Dialog gestartet, in welchem ein zuvor konfigurierter Datenstream gestartet werden kann. Durch die Eingabe „0“ kann auch ein neuer historischer Datenstream konfiguriert werden:



```

1 : Use Local Provider
2 : Connect
3 : Disconnect

10 : Select Provider

20 : Load Config (RecorderSettings)

31 : Start Recording
32 : Stop Recording
33 : Start Historical Stream
34 : Stop Historical Stream
35 : Set Historical Stream Parameter
36 : Is Recording Active

40 : Cancel All Recordings
41 : Cancel All Historical Streams

b : Back to Start
c : Clear Screen
q : Exit Program

[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter a task:~ $ 33
GetHistorical Cmds:
(0) - [Create new GetHistorical cmd]
(1) - "RecordID:29 | Topic:TestSignals/StreamFast"
(2) - "RecordID:39 | Topic:TestSignals/StreamFast"
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please choose id from cmd:~ $ 0

[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter systemID:~ $ cff7975b-b34d-43f7-755d-95cf135f50db
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter topic:~ $ TestSignals/StreamFast
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter symbol layout:~ $ 52a5066f-3c94-d853-f02b-bce62b4a6dea
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter record id:~ $ 1
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter start time in ns:~ $ 13331821883950000
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter end time in ns:~ $ 133318220040054000
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter maxSampleCnt default:~ $ 5000
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter user samplertime default:~ $ -1
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter outputFormat [Bin|Json]:~ $ Bin
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter result topic:~ $ TcBSD_AnalyticsSP_ResultTopic
[TcAnalyticsTest : ASP TcBSD (CX-5AC79C)] Please enter symbol mode [All|Subset]:~ $ All
GetHistoricalStream command for "TcBSD_AnalyticsSP_ResultTopic" send to Provider.

```

Die Parameter definieren dabei eine historisierte Aufnahme, wobei der Parameter **result topic** das MQTT-Topic definiert, an welches die Daten gestreamt werden sollen. Nach der Konfiguration wird automatisch ein Befehl zum Start des Historischen Streams an den Analytics Storage Provider übermittelt.

Durch die Eingabe „35“ in dem Client-Hauptmenü können die Parameter eines aktiven historischen Streams angepasst werden. Der historische Stream wird dabei durch dessen Ergebnis-MQTT-Topic referenziert. Durch die Parameter kann beispielsweise die Geschwindigkeit oder die Paketgröße des Datenstreams bei laufendem Stream angepasst werden. Das Abbrechen eines historischen Datenstreams ist durch die Eingabe „34“ und die Angabe des Ergebnis-MQTT-Topics möglich.

### 6.5.3 Batch Files zur Steuerung

Mit dem Konsolen-Client können Batch-Files zur Steuerung des Analytics Storage Providers erstellt werden. Dazu werden einige Parameter zur Verfügung gestellt:

-Help / -H / -?	Gibt eine Beschreibung von allen Parametern zurück
-----------------	--

Parameter für die Konfigurationseinstellungen:

-CreateASPConfig	Erstelle eine neue Analytics Storage Provider Einstellungs-XML
-MainTopic <mainTopic>	Analytics Storage Provider Main Topic
-Comment <comment>	Analytics Storage Provider Kommentar
-EventLogTrace <True False>	Trace zum Ereignisprotokoll
-DebugLog <True False>	Zusätzlicher DebugLog
-StorageType <type>	Speichertyp ( <b>ANALYTICSFILE, AZURESQ, AZUREBLOB</b> )
-StorageConnString <connString>	Connectionstring oder Pfad zum Speicher
-TlsType <Tls1.0 Tls1.1 Tls1.2>	Tls Typ (für AzureBlob)
-MaxDuration <duration (sec)>	Maximale Dauer eines TAY-File
-MaxWriteLen <writeLen (bytes)>	Maximale Länge eines Datenpakets

## Konfigurationsparameter:

-LocalProvider	Verwende die Verbindungseinstellungen des lokal installierten Analytics Storage Provider
-ConfigFile <path>	Verwende alle Konfigurationen aus der Konfigurationsdatei eines Analytics Storage Provider Recorder Fenster
-ProviderGuid <guid>	Provider des Analytics Storage Providers, der verwendet werden soll
-ConfigCmdID <id>	ID-Nummer der vorkonfigurierten Aufnahme in der Konfigurationsdatei
-ConfigCmdAlias <alias>	Alias von der vorkonfigurierten Aufnahme in der Konfigurationsdatei

## Verbindungsparameter:

-Broker /-Host <hostname>	Hostname oder IP-Adresse des verwendeten Brokers
-Port <port>	Port des Brokers (Standardwert: 1883)
-User <username>	Username für die Verbindung
-Password / -Pwd <password>	Passwort für die Verbindung
-CA <path>	Pfad zum CA-Zertifikat für die Verbindung
-Cert <path>	Pfad zum Zertifikat für die Verbindung
-Key_Cert <path>	Pfad zur Schlüssel-Datei für die Verbindung
-Key_Pwd <password>	Passwort für die Schlüsseldatei für die Verbindung

## Funktionsparameter:

-StartRecord	Sendet ein StartRecord Kommando
-StopRecord	Sendet ein StopRecord Kommando
-IsRecordingActive	Prüft, ob eine Aufnahme aktuell läuft
-GetHistorical	Sendet ein GetHistoricalData Kommando
-StopHistorical	Sendet ein StopHistoricalData Kommando
-UpdateHistorical	Sendet ein HistoricalUpdate Kommando
-CancelAllRec	Sendet ein Cancel Kommando an alle aktiven Aufnahmen
-CancelAllHist	Sendet ein Cancel Kommando an alle aktiven historischen Datenstreams

## Aufnahme Start/Stop Parameter:

-Alias <alias>	Alias Name der Aufnahme
-RecName <record>	Alias Name der Datensatzes
-Topic <topic>	Topic, das aufgenommen werden soll
-DataFormat <Bin Json>	Datenformat des Live-Datastream
-Duration <seconds>	Dauer der Aufnahme
-Ringbuffer <None TimeBased DataBased>	Ringpuffermodus (Standardwert: <b>Default</b> )
-RinbufferPara <minutes/MB>	Parameter für den Ringpuffer (in Sekunden oder Megabyte)
-Mode <All Subset>	Modus der Aufnahme. Nimmt alle Symbole und eine Untermenge der Symbole auf
-Symbols / -Sym <Symbol1,Symbol2>	Liste der Symbol-Untermenge als Komma-separierte Liste
-RecorderGuid <guid>	Guid von dem Analytics Storage Provider Recorder Fenster

**Historische Datenstream Start/Stop Parameter:**

-SystemID <systemID guid>	System ID des aufgenommenen Datensatzes
-Topic <topic>	Topic des aufgenommenen Datensatzes
-Layout <layout guid>	Layout von dem aufgenommenen Datensatz
-RecordID <id>	ID von dem Datensatz, der gestreamt werden soll
-StartTime <time ns>	Startzeitpunkt des Datensatzes, der gestreamt werden soll in Nanosekunden
-EndTime <time ns>	Endzeitpunkt des Datensatzes, der gestreamt werden soll in Nanosekunden
-MaxSamples <samples>	Maximale Anzahl Samples (Standardwert: 5000)
-UsrSampleTime <ms>	Abtastrate. (Standardwert: -1; Abtastrate der Aufnahme)
-DataFormat <Bin Json>	Datenformat des Datenstreams
-ResultTopic <topic>	Ergebnis-MQTT-Topic, auf welches die Daten gestreamt werden
-Mode <All Subset>	Streaming-Modus. Streamt alle oder eine Untermenge der Symbole
-Symbols / -Sym <Symbol1,Symbol2>	Liste der Symbol-Untermenge als Komma-separierte Liste

**Historische Datenstream Update Parameter:**

-MaxSamples <samples>	Maximale Anzahl Samples (Standardwert: 5000)
-UsrSampleTime <ms>	Abtastrate. (Standardwert: -1; Abtastrate der Aufnahme)
-MaxPackSize <samples>	Maximale Nachrichtengröße in Kilobyte
-SendDuration <ms>	Wartezeit zwischen dem Senden von Nachrichten in Millisekunden
-ResultTopic <topic>	Ergebnis-MQTT-Topic, auf welches die Daten gestreamt werden

**Kommandozeilen Beispiele:**

**Konfiguration erstellen:**

```
TwinCAT.Analytics.StorageProvider.Client
  -CreateASPConfig
    -MainTopic Beckhoff/ASPTest
    -Comment Analytics Storage Provider (Test)
    -EventLogTrace False
    -DebugLog False
    -StorageType ANALYTICSFILE
      -StorageConnString C:\TwinCAT\Functions\TF3520-Analytics-StorageProvider\Storage
      -MaxDuration 120
```

```

-MaxWriteLen 2048
-Broker 172.17.62.135
-Port 1883
-User tcanalytics
-Pwd 123

```

### Aufnahme mit lokalem Analytics Storage Provider starten:

```

TwinCAT.Analytics.StorageProvider.Client
-localprovider
-startrecord
  -alias cmdTest
  -recname cmdRec1
  -topic TestSignals/TestStream
  -dataformat Bin
  -Duration 30
  -mode Subset
  -Symbols Variables.fCosine,Variables.fSine

```

### Konfigurationsdatei einer Aufnahme starten:

```

TwinCAT.Analytics.StorageProvider.Client
-ConfigFile "C:
\Users\User\AppData\Roaming\Beckhoff\TwinCAT Analytics Storage Provider\TcAnalyticsStorageProvider_R
ecorder.xml"
-ProviderGuid 76141a7f-e580-4281-99d8-1b8a75ca014d
-startrecord
-ConfigCmdAlias cmdTest

```

### Aufnahmestatus prüfen

```

TwinCAT.Analytics.StorageProvider.Client
-Broker 172.17.62.135
-Port 1883
-User tcanalytics
-Pwd 123
-ProviderGuid 76141a7f-e580-4281-99d8-1b8a75ca014d
-IsRecordingActive
  -alias cmdTest
  -recorderGuid a8e171d2-712d-bd8e-da15-7eef28b71ad2

```

### Alle Aufnahmen stoppen:

```

TwinCAT.Analytics.StorageProvider.Client
-Broker 172.17.62.135
-Port 1883
-User tcanalytics
-Pwd 123
-ProviderGuid 76141a7f-e580-4281-99d8-1b8a75ca014d
-CancelAllRec

```

### Historischen Datenstream starten:

```

TwinCAT.Analytics.StorageProvider.Client
-localprovider
-GetHistorical
  -systemID c29ac2d4-76ce-ff44-4d7f-355ffbcca6bf
  -layout 9a8e171d-712d-bd8e-da15-7eef28b71ad2
  -topic TestSignals/TestStream
  -recordID 1
  -startTime 132696863612730000
  -endTime 13269686417720000
  -maxSamples 5000
  -usrSampleTime -1
  -resultTopic _TestSignals/TestStream/123
  -dataformat Bin
  -mode Subset -symbols Variables.fSine

```

## 7 SPS API

### 7.1 Funktionsbausteine

#### 7.1.1 Topic-Architektur

##### 7.1.1.1 Commands

##### 7.1.1.1.1 T\_ALY\_SPCancel\_Cmd

T_ALY_SPCancel_Cmd	
eCancelType	E_CancelType
arrParameter	ARRAY [0..99] OF T_MaxString

#### Syntax

Definition:

```
FUNCTION_BLOCK T_ALY_SPCancel_Cmd EXTENDS T_ALY_JsonPayload
VAR_INPUT
    eCancelType : E_CancelType;
    arrParameter : ARRAY [0..99] OF T_MaxString;
END_VAR
```

#### Vererbungshierarchie

[T\\_ALY\\_JsonPayload \[▶ 81\]](#)

T\_ALY\_SPCancel\_Cmd

#### 📁 Eingänge

Name	Typ	Beschreibung
eCancelType	E_CancelType <a href="#">[▶ 89]</a>	
arrParameter	ARRAY [0..99] OF T_MaxString	

#### 📁 Methoden

Name	Definitionort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.1.2 T\_ALY\_SPGetHistorical\_Cmd

T_ALY_SPGetHistorical_Cmd	
—sTopic	T_MaxString
—sLayout	GUID
—eMode	E_SymbolMode
—eOutputFormat	E_RawDataFormat
—nMaxSampleCount	UDINT
—nUserSampleTime	DINT
—nRecordID	DINT
—nStartTimestamp	LINT
—nEndTimestamp	LINT
—sResultTopic	T_MaxString
—arrSymbols	ARRAY [0..255] OF T_ALY_Symbol

#### Syntax

##### Definition:




```
FUNCTION_BLOCK T_ALY_SPGetHistorical_Cmd EXTENDS T_ALY_JsonPayload
VAR_INPUT
    sTopic      : T_MaxString;
    sLayout     : GUID;
    eMode       : E_SymbolMode := E_SymbolMode.All;
    eOutputFormat : E_RawDataFormat := E_RawDataFormat.Bin;
    nMaxSampleCount : UDINT := 3000;
    nUserSampleTime : DINT := -1;
    nRecordID   : DINT;
    nStartTimestamp : LINT;
    nEndTimestamp  : LINT;
    sResultTopic : T_MaxString;
    arrSymbol    : ARRAY [0..255] OF T_ALY_Symbol;
END_VAR
```

#### Vererbungshierarchie





[T\\_ALY\\_JsonPayload](#) [► 81]

T\_ALY\_SPGetHistorical\_Cmd

 Eingänge

Name	Typ	Beschreibung
sTopic	T_MaxString	Topic-Name des aufgezeichneten Live-Streams.
sLayout	GUID	Layout-GUID der Aufzeichnung.
eMode	<a href="#">E_SymbolMode</a> 	Abrufen aller Symbole oder nur einer Teilmenge.
eOutputFormat	<a href="#">E_RawDataFormat</a> 	Format der zurückgegebenen Daten (derzeit wird nur „Bin“ unterstützt).
nMaxSampleCount	UDINT	Maximale Anzahl von Samples in einem Nutzlastpaket.
nUserSampleTime	DINT	Samplezeit in Millisekunden des zurückgegebenen Streams. (-1 verwendet die aufgezeichnete Samplezeit.)
nRecordID	DINT	Nummer der Aufzeichnung.
nStartTimestamp	LINT	Startzeit
nEndTimestamp	LINT	Endzeit
sResultTopic	T_MaxString	Topic-Name des Ergebnis-Streams.
arrSymbol	ARRAY [0..255] OF <a href="#">T_ALY_Symbol</a> 	Wenn der SymbolMode Subset ist, wird nur die Liste dieser Symbole zurückgegeben.

 Methoden

Name	Definitionsart	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload</a> 	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload</a> 	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload</a> 	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload</a> 	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.1.3 T\_ALY\_SPReadStreamRecords\_Cmd



#### Syntax

Definition:

```

FUNCTION_BLOCK T_ALY_SPReadStreamRecords_Cmd EXTENDS T_ALY_JsonPayload
VAR_INPUT
    sStreamTopic : STRING(255);
    sStreamSystemID : GUID;
    sStreamLayout : GUID;
    nRecordStartIndex : DINT;
    nMaxRecordCount : DINT;
    sResultTopic : T_MaxString;
END_VAR

```

#### Vererbungshierarchie

[T\\_ALY\\_JsonPayload \[► 81\]](#)

T\_ALY\_SPReadStreamRecords\_Cmd

#### 📌 Eingänge

Name	Typ	Beschreibung
sStreamTopic	STRING(255)	Topic-Name des aufgezeichneten Live-Streams.
sStreamSystemID	GUID	SystemID des Target Systems von wo der Live-Stream gesendet wurde
sStreamLayout	GUID	Layout-GUID der Aufzeichnung
nRecordStartIndex	DINT	Startindex der ersten Aufzeichnung die gelesen werden soll.
nMaxRecordCount	DINT	Gesamtanzahl der Aufzeichnungen, die gelesen werden soll.
sResultTopic	T_MaxString	Topic-Name des Ergebnis-Streams



Methoden

Name	Definitionsort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.1.4 T\_ALY\_SPRecordData\_Cmd

```

T_ALY_SPRecordData_Cmd
—sAlias T_MaxString
—sRecordName T_MaxString
—eRecording E_RecordMode
—sRecorder GUID
—sRecorderAlias T_MaxString
—sTopic T_MaxString
—eDataFormat E_RawDataFormat
—nDuration DINT
—eRingBufferMode E_RingBufferMode
—nRingBufferParameter DINT
—eMode E_SymbolMode
—sSymbolLayout GUID
—arrSymbols ARRAY [0..255] OF T_ALY_Symbol
    
```

#### Syntax

Definition:

```

FUNCTION_BLOCK T_ALY_SPRecordData_Cmd EXTENDS T_ALY_JsonPayload
VAR_INPUT
    sAlias : T_MaxString;
    sRecordName : T_MaxString;
    eRecording : E_RecordMode;
    sRecorder : GUID;
    sRecorderAlias : T_MaxString;
    sTopic: T_MaxString;
    eDataFormat : E_RawDataFormat;
    nDuration : DINT;
    eRingBufferMode : E_RingBufferMode;
    nRingBufferParameter : DINT;
    eMode : E_SymbolMode;
    sSymbolLayout : GUID;
    arrSymbols : ARRAY [0..255] OF T_ALY_Symbol;
END_VAR
    
```

#### Vererbungshierarchie

[T\\_ALY\\_JsonPayload](#) [▶ 81]

T\_ALY\_SPRecordData\_Cmd

## 🚪 Eingänge

Name	Typ	Beschreibung
sAlias	T_MaxString	Alias-Name für die Aufzeichnung.
sRecordName	T_MaxString	Name für diese Aufzeichnung.
eRecording	E_RecordMode [▶ 90]	Start oder Stopp der Aufzeichnung.
sRecorder	GUID	Individuelle GUID des Recorders.
sRecorderAlias	T_MaxString	Alias-Name für den Recorder.
sTopic	T_MaxString	Topic-Name des Live-Streams.
eDataFormat	E_RawDataFormat [▶ 89]	Speichern des Datenformats (derzeit wird nur das Binärformat unterstützt).
nDuration	DINT	Dauer der Aufzeichnung in Minuten. (-1 unbegrenzt)
eRingBufferMode	E_RingBufferMode [▶ 90]	Ringpuffermodus
nRingBufferParameter	DINT	TimeBased => Parameter in Minuten. DataBased => Parameter in Megabyte.
eMode	E_SymbolMode [▶ 91]	Aufzeichnen aller Symbole oder nur einer Teilmenge.
sSymbolLayout	GUID	
arrSymbols	ARRAY [0..255] OF T_ALY_Symbol [▶ 79]	Wenn der SymbolMode Subset ist, wird nur die Liste dieser Symbole aufgezeichnet.

## 🔧 Methoden

Name	Definitionsart	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von T_ALY_JsonPayload [▶ 81]	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von T_ALY_JsonPayload [▶ 81]	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von T_ALY_JsonPayload [▶ 81]	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von T_ALY_JsonPayload [▶ 81]	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.1.5 T\_ALY\_SPReloadHistoricalStreams\_Cmd

T_ALY_SPReloadHistoricalStreams_Cmd	
eReloadType	E_ReloadType
arrParameter	ARRAY [0..9] OF ARRAY [0..1] OF T_MaxString

**Syntax**

Definition:

```
FUNCTION_BLOCK T_ALY_SPREloadHistoricalStreams_Cmd EXTENDS T_ALY_JsonPayload
VAR_INPUT
    eReloadType : E_ReloadType;
    arrParameter : ARRAY [0..9] OF ARRAY [0..1] OF T_MaxString;
END_VAR
```

**Vererbungshierarchie**

[T\\_ALY\\_JsonPayload \[▶ 81\]](#)

T\_ALY\_SPREloadHistoricalStreams\_Cmd

**📌 Eingänge**

Name	Typ	Beschreibung
eReloadType	<a href="#">E_ReloadType [▶ 90]</a>	Auswahl des Aktualisierungsmodus
arrParameter	ARRAY [0..9] OF ARRAY [0..1] OF T_MaxString	Zusatzparameter

**🔗 Methoden**

Name	Definitionsort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload [▶ 81]</a>	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

**7.1.1.1.6 T\_ALY\_SPSetGetHistoricalDataState\_Cmd**

**T\_ALY\_SPSetGetHistoricalDataState\_Cmd**

- sResultTopic *T\_MaxString*
- eState *E\_SetGetHistoricalDataState*
- nSendDuration\_ms *DINT*
- nRestartTimestamp *LINT*
- nMaxSampleCount *UDINT*
- nMaxPackageSize\_KB *DINT*
- nUserSampleTime *LINT*

**Syntax**

Definition:

```

FUNCTION_BLOCK T_ALY_SPSetGetHistoricalDataState_Cmd EXTENDS T_ALY_JsonPayload
VAR_INPUT
    sResultTopic : T_MaxString;
    eState : E_SetGetHistoricalDataState;
    nSendDuration_ms : DINT;
    nRestartTimestamp : LINT;
    nMaxSampleCount : UDINT;
    nMaxPackageSize_KB: DINT;
    nUserSampleTime : LINT;
END_VAR

```

## Vererbungshierarchie

T\_ALY\_JsonPayload [[▶ 81](#)]

T\_ALY\_SPSetGetHistoricalDataState\_Cmd

### Eingänge

Name	Typ	Beschreibung
sResultTopic	T_MaxString	Topic-Name des Ergebnis-Streams (wird wie ein Handle verwendet).
eState	<a href="#">E_SetGetHistoricalDataState ▶ 91</a>	Status des Historischen-Streams
nSendDuration_ms	DINT	Wartezeit zwischen dem versenden der einzelnen Paketen
nRestartTimestamp	LINT	Zeitstempel an dem der Ergebnis-Stream fortgeführt wird.
nMaxSampleCount	UDINT	Maximale Anzahl an Einträgen in einem Paket
nMaxPackageSize_KB	DINT	Maximal Größe eines Pakets
nUserSampleTime	LINT	Samplezeit in Millisekunden des zurückgegebenen Streams (-1 verwendet die aufgezeichnete Samplezeit).

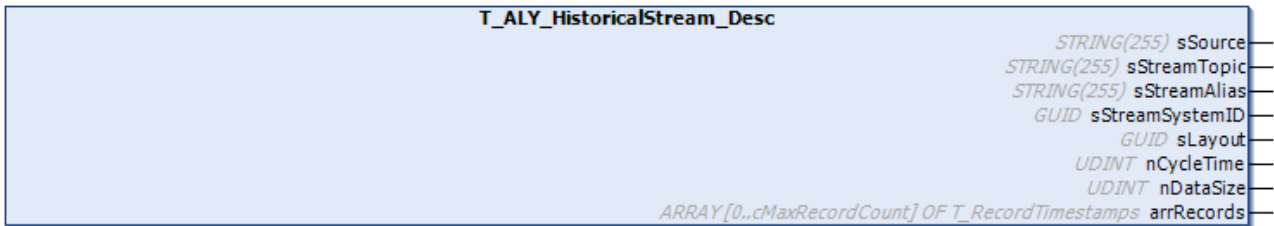
### Methoden

Name	Definitionsart	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload ▶ 81</a>	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload ▶ 81</a>	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload ▶ 81</a>	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload ▶ 81</a>	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

7.1.1.2 Descriptions

7.1.1.2.1 T\_ALY\_HistoricalStream\_Desc



Syntax

Definition:

```

FUNCTION_BLOCK T_ALY_HistoricalStream_Desc EXTENDS T_ALY_JsonPayload
VAR_OUTPUT
    sSource : STRING(255);
    sStreamTopic : STRING(255);
    sStreamAlias : STRING(255);
    sStreamSystemID : GUID;
    sLayout : GUID;
    nCycleTime: UDINT;
    nDataSize : UDINT;
    arrRecords : ARRAY [0..cMaxRecordCount] OF T_RecordTimestamps;
END_VAR
    
```

Vererbungshierarchie

T\_ALY\_JsonPayload [► 81]

T\_ALY\_HistoricalStream\_Desc

Ausgänge

Name	Typ	Beschreibung
sSource	STRING(255)	Name der Datenquelle
sStreamTopic	STRING(255)	Topic-Name des aufgezeichneten Streams
sStreamAlias	STRING(255)	Alias-Name des Streams
sStreamSystemID	GUID	SystemID-GUID des Streams
sLayout	GUID	Layout-GUID der Aufzeichnung
nCycleTime	UDINT	Zykluszeit der Aufzeichnung
nDataSize	UDINT	Datengröße eines Eintrags der Aufzeichnung
arrRecords	ARRAY [0..cMaxRecordCount] OF T_RecordTimestamps [► 80]	Zeitstempel der verschiedenen Aufzeichnungen

## Methoden

Name	Definitionsort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.3 Info

#### 7.1.1.3.1 T\_ALY\_ReadStreamRecord\_Info

T_ALY_ReadStreamRecord_Info	
	<i>DINT</i> nRecordCountAll
	<i>DINT</i> nRecordCount
	<i>ARRAY [0..cMaxRecordCount] OF T_RecordTimestamps</i> arrRecords

### Syntax

Definition:

```
FUNCTION_BLOCK T_ALY_HistoricalStream_Desc EXTENDS T_ALY_JsonPayload
VAR_OUTPUT
    nRecordCountAll: UDINT;
    nRecordCount : UDINT;
    arrRecords : ARRAY [0..cMaxRecordCount] OF T_RecordTimestamps;
END_VAR
```

### Vererbungshierarchie

[T\\_ALY\\_JsonPayload](#) [▶ 81]

T\_ALY\_ReadStreamRecord\_Info

## Ausgänge

Name	Typ	Beschreibung
nRecordCountAll	UDINT	Anzahl aller vorhanden Aufzeichnungen
nRecordCount	UDINT	Anzahl ausgelesener Aufzeichnungen
arrRecords	<a href="#">ARRAY [0..cMaxRecordCount] OF T_RecordTimestamps</a> [▶ 80]	Zeitstempel der ausgelesenen Aufzeichnungen

Methoden

Name	Definitionsort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.4 SubTypes

#### 7.1.1.4.1 T\_ALY\_Symbol



#### Syntax

Definition:

```

FUNCTION_BLOCK T_ALY_Symbol EXTENDS T_ALY_JsonPayload
VAR_INPUT
    sName : T_MaxString;
    sBaseType : T_MaxString;
    nBitOffset : UDINT;
    nBitSize : UDINT;
END_VAR
    
```

#### Vererbungshierarchie

[T\\_ALY\\_JsonPayload](#) [▶ 81]

T\_ALY\_Symbol

#### Eingänge

Name	Typ	Beschreibung
sName	T_MaxString	Name des Symbols
sBaseType	T_MaxString	Datentyp des Symbols
nBitOffset	UDINT	Bit-Offset des Symbols
nBitSize	UDINT	Bitgröße des Symbols

## Methoden

Name	Definitionsort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.4.2 T\_RecordTimestamps



## Syntax

### Definition:

```
FUNCTION_BLOCK T_RecordTimestamps EXTENDS T_ALY_JsonPayload
VAR_OUTPUT
    nRecordID : DINT;
    sAlias    : STRING(255);
    nStartTimestamp : LINT;
    nEndTimestamp : LINT;
END_VAR
```

## Vererbungshierarchie

[T\\_ALY\\_JsonPayload](#) [▶ 81]

T\_RecordTimestamps

## Ausgänge

Name	Typ	Beschreibung
nRecordID	DINT	Nummer der Aufzeichnung
sAlias	STRING(255)	Alias-Name der Aufzeichnung
nStartTimestamp	LINT	Start-Zeitstempel der Aufzeichnung
nEndTimestamp	LINT	End-Zeitstempel der Aufzeichnung



☞ Methoden

Name	Definitionsort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

### 7.1.1.5 Base Types

#### 7.1.1.5.1 T\_ALY\_JsonPayload



#### Syntax

Definition:

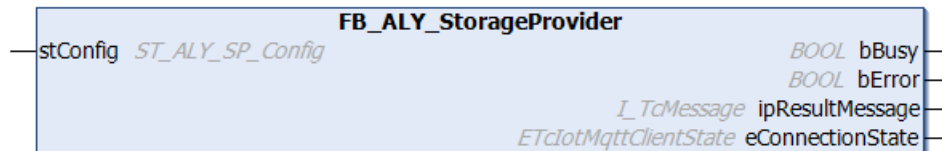
```
FUNCTION_BLOCK INTERNAL T_ALY_JsonPayload
```

☞ Methoden

Name	Definitionsort	Beschreibung
Reset		Zurücksetzen aller Werte im Nutzlast-FB
Init_JsonValue	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-Objekt
Init_String	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Initialisierung des FB mit Json-String
GetJsonLength	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Länge der Json-Nutzlast
GetJsonString	Vererbt von <a href="#">T_ALY_JsonPayload</a> [▶ 81]	Abrufen der Json-Nutzlast als String

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

## 7.1.2 FB\_ALY\_StorageProvider



Der FB\_ALY\_StorageProvider ist ein Client-FB für die Kommunikation mit einer Storage Provider-Instanz. Der FB bietet Methoden, um historische Daten auszulösen oder Aufzeichnungen zu starten/stoppen.

### Syntax

#### Definition:

```

FUNCTION_BLOCK FB_ALY_StorageProvider
VAR_INPUT
    stConfig : ST_ALY_SP_Config;
END_VAR
VAR_OUTPUT
    bBusy : BOOL;
    bError : BOOL;
    ipResultMessage : I_TcMessage;
    eConnectionState : ETcIotMqttClientState;
END_VAR

```

#### Eingänge

Name	Typ	Beschreibung
stConfig	ST_ALY_SP_Config [ <a href="#">▶ 88</a> ]	Struktur für die Konfiguration des FB.

#### Ausgänge

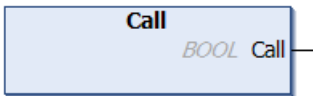
Name	Typ	Beschreibung
bBusy	BOOL	TRUE, sobald eine Methode des Funktionsbausteins aktiv ist.
bError	BOOL	Wird TRUE, sobald eine Fehlersituation auftritt.
ipResultMessage	I_TcMessage	Nachrichtenschnittstelle des TwinCAT 3 EventLoggers, die nähere Informationen über den Rückgabewert bietet.
eConnectionState	ETcIotMqttClientState	Gibt den Status der Verbindung zwischen Client und Broker als Aufzählung ETcIotMqttClientState an.

Methoden

Name	Definitionsort	Beschreibung
<a href="#">Call</a> [ <a href="#">▶ 83</a> ]	Local	Methode für Hintergrundkommunikation mit dem TwinCAT-Treiber. Die Methode muss zyklisch aufgerufen werden.
<a href="#">Cancel</a> [ <a href="#">▶ 83</a> ]	Local	Methode für den Abbruch von Aktivitäten des TwinCAT Analytics Storage Providers.
<a href="#">GetHistoricalData</a> [ <a href="#">▶ 84</a> ]	Local	Methode zum Anfordern historischer Daten.
<a href="#">ReadHistoricalStreams</a> [ <a href="#">▶ 84</a> ]	Local	Methode zum Auslesen aller historischer Streams.
<a href="#">ReadStreamRecords</a> [ <a href="#">▶ 85</a> ]	Local	Methode zum auslesen aller Aufnahmen eines historischen Streams
<a href="#">ResetCommunication</a> [ <a href="#">▶ 86</a> ]	Local	Methode zum zurücksetzen der MQTT Verbindung zum Broker.
<a href="#">SendCommand</a> [ <a href="#">▶ 86</a> ]	Local	Generische Methode zum absenden verschiedener Kommandos.
<a href="#">SetHistoricalDataState</a> [ <a href="#">▶ 86</a> ]	Local	Methode zum setzen verschiedener Parameter eines historischen Streams
<a href="#">StartRecord</a> [ <a href="#">▶ 87</a> ]	Local	Startet die Aufzeichnung eines Live-MQTT-Binärstroms.
<a href="#">StopRecord</a> [ <a href="#">▶ 87</a> ]	Local	Stoppt die ausgewählte Aufzeichnung.

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v3.1.4022.25	PC oder CX (x64, x86, ARM)	Tc3_AnalyticsStorageProvider

7.1.2.1 Call



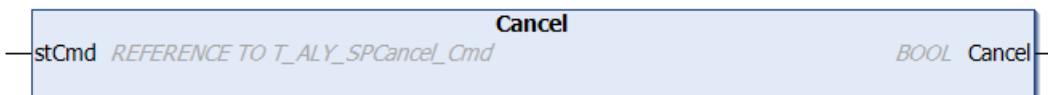
Syntax

```
METHOD Call : BOOL
```

Rückgabewert

Name	Typ	Beschreibung
Call	BOOL	

7.1.2.2 Cancel



Syntax

```
METHOD Cancel : BOOL
VAR_INPUT
    stCmd : REFERENCE TO T_ALY_SPCancel_Cmd;
END_VAR
```

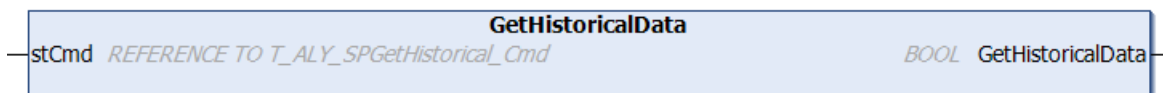
### 📌 Eingänge

Name	Typ	Beschreibung
stCmd	REFERENCE TO <a href="#">T_ALY_SPCancel_Cmd [▶ 69]</a>	Json-Befehl zum Abbruch der Operationen des TwinCAT Analytics Storage Providers

### 📌 Rückgabewert

Name	Typ	Beschreibung
Cancel	BOOL	Ist TRUE, wenn abgeschlossen

## 7.1.2.3 GetHistoricalData



### Syntax

```
METHOD GetHistoricalData : BOOL
VAR_INPUT
    stCmd : REFERENCE TO T_ALY_SPHistorical_Cmd;
END_VAR
```

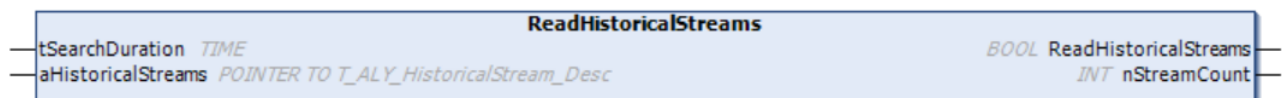
### 📌 Eingänge

Name	Typ	Beschreibung
stCmd	REFERENCE TO <a href="#">T_ALY_SPGetHistorical_Cmd [▶ 70]</a>	Json-Befehl zum Abrufen historischer Daten von TwinCAT Analytics Storage Provider

### 📌 Rückgabewert

Name	Typ	Beschreibung
GetHistoricalData	BOOL	Ist TRUE, wenn abgeschlossen

## 7.1.2.4 ReadHistoricalStreams



### Syntax

```
METHOD GetHistoricalData : BOOL
VAR_INPUT
    tSearchDuration : TIME := TIME#5s0ms
    aHistoricalStreams : POINTER TO T_ALY_HistoricalStream_Desc;
    nStreamCount : INT
END_VAR
```

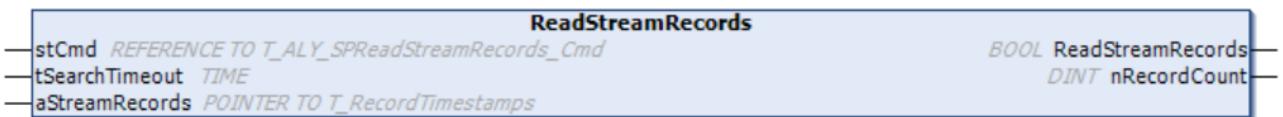
 **Eingänge**

Name	Typ	Beschreibung
tSearchDuration	TIME	Zeitspanne in der auf Rückmeldung gewartet wird.
aHistoricalStreams	POINTER TO T_ALY HistoricalStream Desc <a href="#">▶ 77</a>	Beschreibung der verschieden historischen Streams

 **Rückgabewert**

Name	Typ	Beschreibung
ReadHistoricalStreams	BOOL	Ist TRUE, wenn abgeschlossen
nStreamCount	INT	Anzahl der ausgelesenen Streams

### 7.1.2.5 ReadStreamRecords



**Syntax**

```

METHOD ReadStreamRecords : BOOL
VAR_INPUT
    stCmd : REFERENCE TO T_ALY_SPCmd;
    tSearchTimeout : TIME := TIME#5s0ms;
    aStreamRecords : POINTER TO T_RecordTimestamps;
    nRecordCount : DINT;
END_VAR
    
```

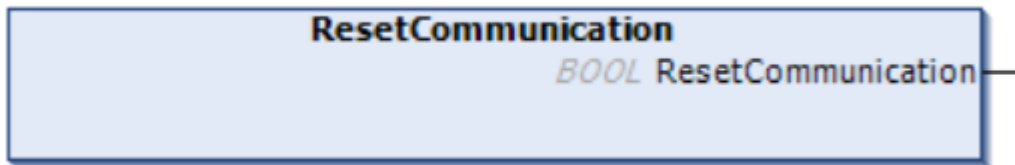
 **Eingänge**

Name	Typ	Beschreibung
stCmd	REFERENCE TO T_ALY_SPCmd <a href="#">▶ 72</a>	Json-Befehl zum Abrufen Aufnahmen eines historisches Streams von TwinCAT Analytics Storage Provider
tSearchTimeout	TIME	Wartezeit auf die Antwort
aStreamRecords	POINTER TO T_RecordTimestamps <a href="#">▶ 80</a>	Ausgelesene Aufnahmen

 **Rückgabewert**

Name	Typ	Beschreibung
ReadStreamRecords	BOOL	Ist TRUE, wenn abgeschlossen
nRecordCount	DINT	Anzahl der ausgelesenen Aufnahmen

### 7.1.2.6 ResetCommunication



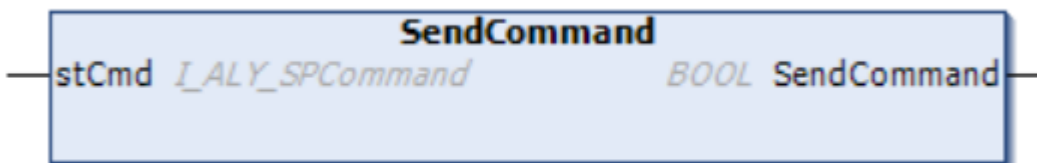
#### Syntax

```
METHOD ResetCommunication : BOOL
VAR_INPUT
END_VAR
```

#### 📌 Rückgabewert

Name	Typ	Beschreibung
ResetCommunication	BOOL	Ist TRUE, wenn abgeschlossen

### 7.1.2.7 SendCommand



#### Syntax

```
METHOD SendCommand : BOOL
VAR_INPUT
    stCmd : I_ALY_SPCCommand;
END_VAR
```

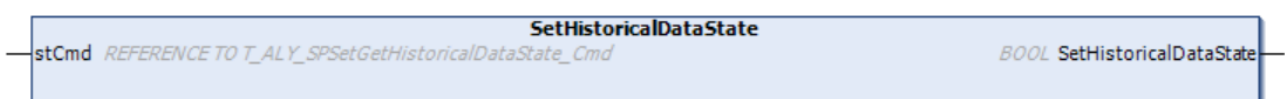
#### 📌 Eingänge

Name	Typ	Beschreibung
stCmd	I_ALY_SPCCommand	Json-Befehl zum interagieren mit dem TwinCAT Analytics Storage Provider

#### 📌 Rückgabewert

Name	Typ	Beschreibung
SendCommand	BOOL	Ist TRUE, wenn abgeschlossen

### 7.1.2.8 SetHistoricalDataState



**Syntax**

```
METHOD SetHistoricalDataState : BOOL
VAR_INPUT
    stCmd : REFERENCE TO T_ALY_SPGetSetHistoricalDataState_Cmd;
END_VAR
```

 **Eingänge**

Name	Typ	Beschreibung
stCmd	REFERENCE TO T_ALY_SPGetSetHistoricalDataState _Cmd [ <a href="#">▶ 75</a> ]	Json-Befehl zum Setzen von Parametern eines gestarteten historischem Stream vom TwinCAT Analytics Storage Provider

 **Rückgabewert**

Name	Typ	Beschreibung
SetHistoricalDataState	BOOL	Ist TRUE, wenn abgeschlossen

**7.1.2.9 StartRecord**



**Syntax**

```
METHOD StartRecord : BOOL
VAR_INPUT
    stCmd : REFERENCE TO T_ALY_SPRecordData_Cmd;
END_VAR
```

 **Eingänge**

Name	Typ	Beschreibung
stCmd	REFERENCE TO T_ALY_SPRecordData_Cmd [ <a href="#">▶ 73</a> ]	Json-Befehl zum Starten der Aufzeichnung eines Live-Streams

 **Rückgabewert**

Name	Typ	Beschreibung
StartRecord	BOOL	Ist TRUE, wenn abgeschlossen

**7.1.2.10 StopRecord**



**Syntax**

```
METHOD StopRecord : BOOL
VAR_INPUT
    stCmd : REFERENCE TO T_ALY_SPRecordData_Cmd;
END_VAR
```

### Eingänge

Name	Typ	Beschreibung
stCmd	REFERENCE TO T_ALY_SPRecordData_Cmd [ <a href="#">▶_73</a> ]	Json-Befehl zum Stoppen der Aufzeichnung eines Live-Streams

### Rückgabewert

Name	Typ	Beschreibung
StopRecord	BOOL	Ist TRUE, wenn abgeschlossen

## 7.2 Datentypen

### 7.2.1 ST\_ConnectionSettings

#### Syntax

Definition:

```

TYPE ST_ConnectionSettings :
STRUCT
  sHostName      : T_MaxString;
  nHostPort     : UINT := 1883;
  sUserId       : T_MaxString;
  sPassword     : T_MaxString;
  bWithCertificate : BOOL := BOOL;
  sCA           : T_MaxString;
  sCert        : T_MaxString;
  sKey         : T_MaxString;
sKeyPwd : T_MaxString;
END_STRUCT
END_TYPE

```

#### Parameter

Name	Typ	Beschreibung
sHostName	T_MaxString	sHostName kann als Name oder als IP-Adresse festgelegt werden. Wenn keine Angaben gemacht werden, wird der lokale Host verwendet.
nHostPort	UINT	Hier kann der Hostport festgelegt werden: Der Standardwert ist 1883.
sUserId	T_MaxString	Optional kann ein Benutzername angegeben werden.
sPassword	T_MaxString	Hier kann ein Passwort für den Benutzernamen eingegeben werden.
bWithCertificate	BOOL	Falls TRUE, werden die Zertifikate für die Kommunikation verwendet.
sCA	T_MaxString	Zertifikat der Zertifizierungsstelle (CA).
sCert	T_MaxString	Client-Zertifikat für die Authentifizierung beim Broker.
sKey	T_MaxString	Privater Schlüssel des Clients.
sKeyPwd	T_MaxString	Passwort des privaten Schlüssels, falls zutreffend.

### 7.2.2 ST\_ALY\_SP\_Config

#### Syntax

Definition:



```

TYPE ST_Msg :
STRUCT
    sMainTopic      : T_MaxString;
    sProviderGuid   : GUID;
    stConnSettings  : ST_ConnectionSettings
END_STRUCT
END_TYPE
    
```

**Parameter**

Name	Typ	Beschreibung
sMainTopic	T_MaxString	Das Main-Topic, wenn sich der TwinCAT Analytics Storage Provider auf dem Message-Broker befindet.
sProviderGuid	GUID	Die individuelle GUID der TwinCAT Analytics Storage Provider-Instanz.
stConnSettings	<a href="#">ST_ConnectionSettings [► 88]</a>	MQTT-Verbindungseinstellungen für die Verbindung mit dem Message-Broker.

### 7.2.3 E\_CancelType

**Syntax**

Definition:

```

TYPE E_CancelType :
(
    HistoricalData := 0,
    AllRecordData
) INT;
END_TYPE
    
```

**Parameter**

Name	Beschreibung
HistoricalData	Bricht den ausgewählten historischen Datenstrom ab.
AllRecordData	Bricht alle laufenden Aufzeichnungen ab.

### 7.2.4 E\_RawDataFormat

**Syntax**

Definition:

```

TYPE E_RawDataFormat :
(
    Bin := 0,
    Json
) INT;
END_TYPE
    
```

**Parameter**

Name	Beschreibung
Bin	Analytics Binary Stream Format.
Json	TwinCAT Json-Format (wird derzeit nicht unterstützt).

## 7.2.5 E\_RecordMode

### Syntax

Definition:

```
TYPE E_RecordMode :
(
    Start := 0,
    Stop
) INT;
END_TYPE
```

### Parameter

Name	Beschreibung
Start	Startet die konfigurierte Aufzeichnung.
Stop	Stoppt die Aufzeichnung.

## 7.2.6 E\_ReloadType

### Syntax

Definition:

```
TYPE E_ReloadType :
(
    All := 0,
    Specific
) INT;
END_TYPE
```

### Parameter

Name	Beschreibung
All	Es werden alle Aufzeichnungen neu eingelesen.
Specific	Es wird nur eine spezifische Aufnahme neu eingelesen.

## 7.2.7 E\_RingBufferMode

### Syntax

Definition:

```
TYPE E_RingBufferMode:
(
    None := 0,
    TimeBased,
    DataBased
) INT;
END_TYPE
```

### Parameter

Name	Beschreibung
None	Aufzeichnung ohne Ringpuffermodus.
TimeBased	Ringpuffer basiert auf einem bestimmten Zeitraum.
DataBased	Ringbuffer basiert auf einer bestimmten maximalen Datengröße.

## 7.2.8 E\_SetGetHistoricalDataState

### Syntax

Definition:

```
TYPE E_SetGetHistoricalDataState :
(
    Pause,
    Continue_,
    Restart,
    Stop,
    Update
) INT;
END_TYPE
```

### Parameter

Name	Beschreibung
Pause	Abspielen der Aufzeichnung wird pausiert.
Continue_	Abspielen der Aufzeichnung wird fortgesetzt.
Restart	Abspielen der Aufzeichnung wird neu gestartet.
Stop	Abspielen der Aufzeichnung wird beendet.
Update	Parameter zum Abspielen der Aufzeichnung werden aktualisiert.

## 7.2.9 E\_SymbolMode

### Syntax

Definition:

```
TYPE E_SymbolMode :
(
    All := 0,
    Subset
) INT;
END_TYPE
```

### Parameter

Name	Beschreibung
All	Alle Symbole des Streams werden verwendet.
Subset	Nur eine Teilmenge der Symbole wird verwendet.

## 8 Beispiele

### 8.1 SPS Client

Dieses SPS Beispiel zeigt die Verwendung der TwinCAT Analytics Storage Provider Bibliothek. Der Beispiel-Code zeigt das Lesen und Schreiben. Damit das Beispiel in sich geschlossen funktioniert, wird sowohl die Verwendung des Analytics Loggers für das Schicken von Messdaten an einen MQTT Message Broker gezeigt wie auch das Einspielen der historischen Daten über den Analytics Stream Helper.

Grundlage ist ein entsprechend aufgesetzter nativer MQTT Message Broker und ein Analytics Storage Provider Service.

Das SPS Beispiel zeigt folgende Ablaufschritte:

1. Analytics Logger: Stream von Variablen aus einer Globalen Variablen Liste an einen MQTT Message Broker.
2. Read Stream Description: Empfang der Stream Beschreibung vom MQTT Message Broker. Auswertung der JSON Beschreibung für die Verwendung der notwendigen IDs.
3. Analytics Storage Provider: Starten und stoppen von Aufnahmen, sowie Lesen von Aufnahmen und Lesen der historischen Daten.
4. Analytics Stream Helper: Empfangen der historischen Daten vom Analytics Storage Provider und das Mappen der Daten in eine Globale Variablen Liste für die historischen Daten.

Für die Verwendung des Storage Providers in der SPS sind verschiedene GUIDs für die Identifikation von Services und Daten notwendig. Anhand der folgenden Screenshots wird gezeigt, wo die entsprechenden GUIDs gefunden werden können.

- TwinCAT System ID
- Symbol Info GUID
- Storage Provider Recorder GUID
- Storage Provider Service GUID

#### TwinCAT System ID

**About TwinCAT System**

TwinCAT System Service v3.1.0.2434 OK

**TwinCAT** **v3.1.4024.22**

Copyright BECKHOFF Automation © 1996-2021

AMS Net Id:

Logon User:  HW Platform:

User Group:  **System Id:**

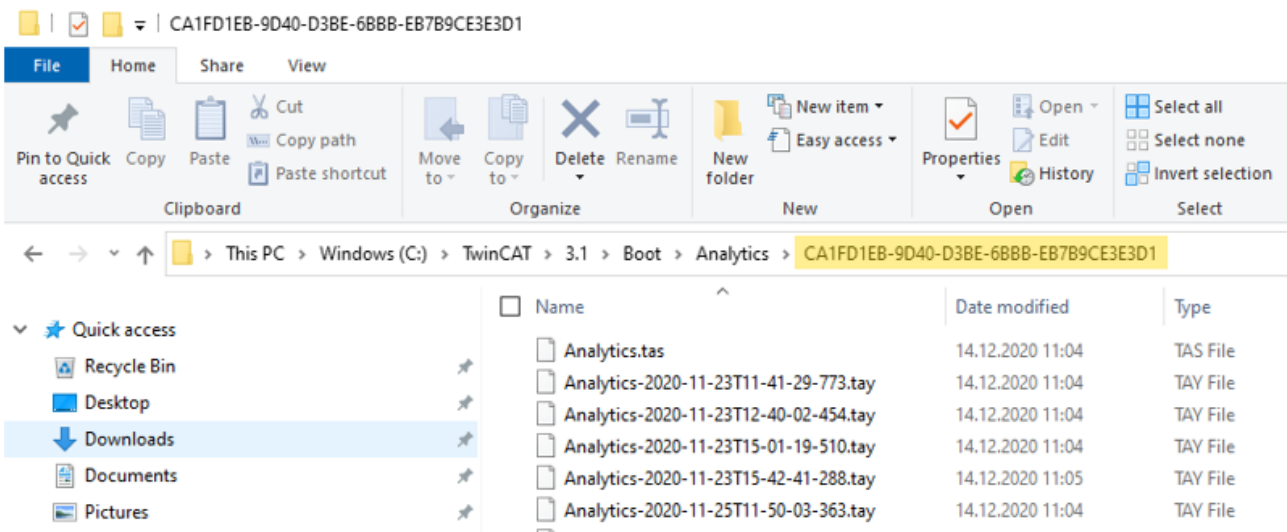
Self Signed Certificate - Fingerprint:

Licenses:

Order No	License	Instances	Status
	TC3 Analytics - Math	cpu license	valid
	TE3500 Analytics Workbench Ju...	cpu license	valid
	TC3 Analytics - IoT	cpu license	valid
	TC3 MMS	cpu license	valid
	TC3 Analytics - XY Path Analysis	cpu license	valid
	TC3 Condition Monitoring Base...	cpu license	valid

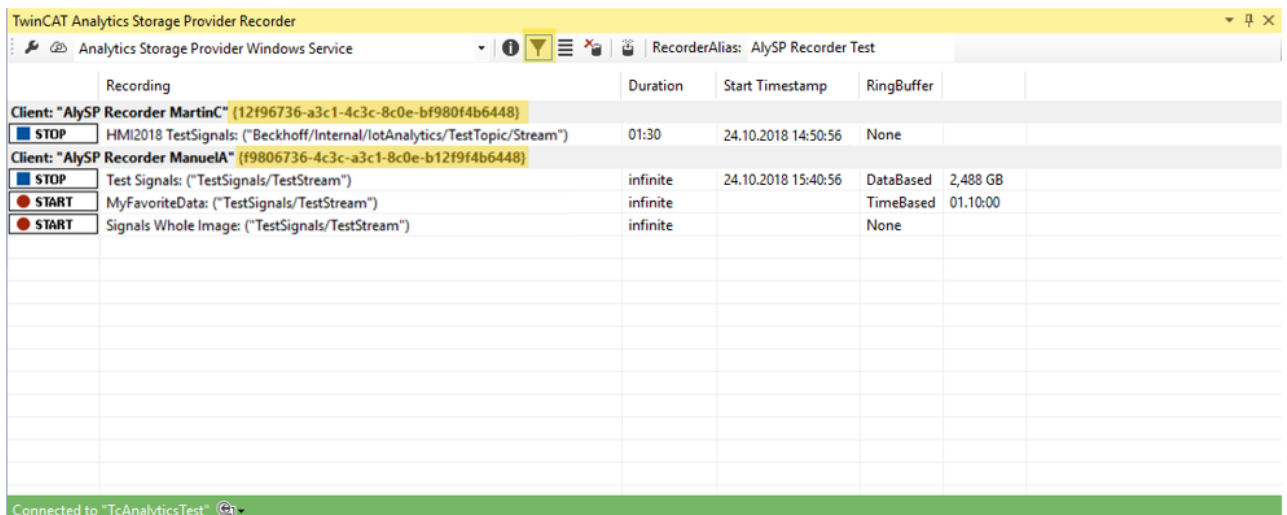
### Symbol Info GUID

Identification ID für die Symbole aus dem TwinCAT Analytics Binär-Format.



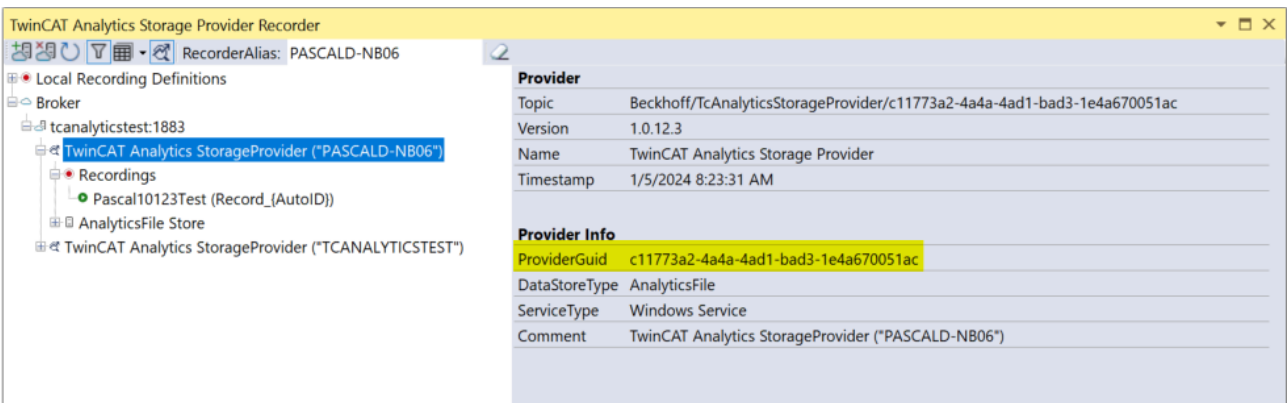
### Storage Provider Recorder GUID

Identification GUID für den Storage Provider Recorder im Visual Studio® - dieser wird mit dem TwinCAT Measurement Setup installiert.



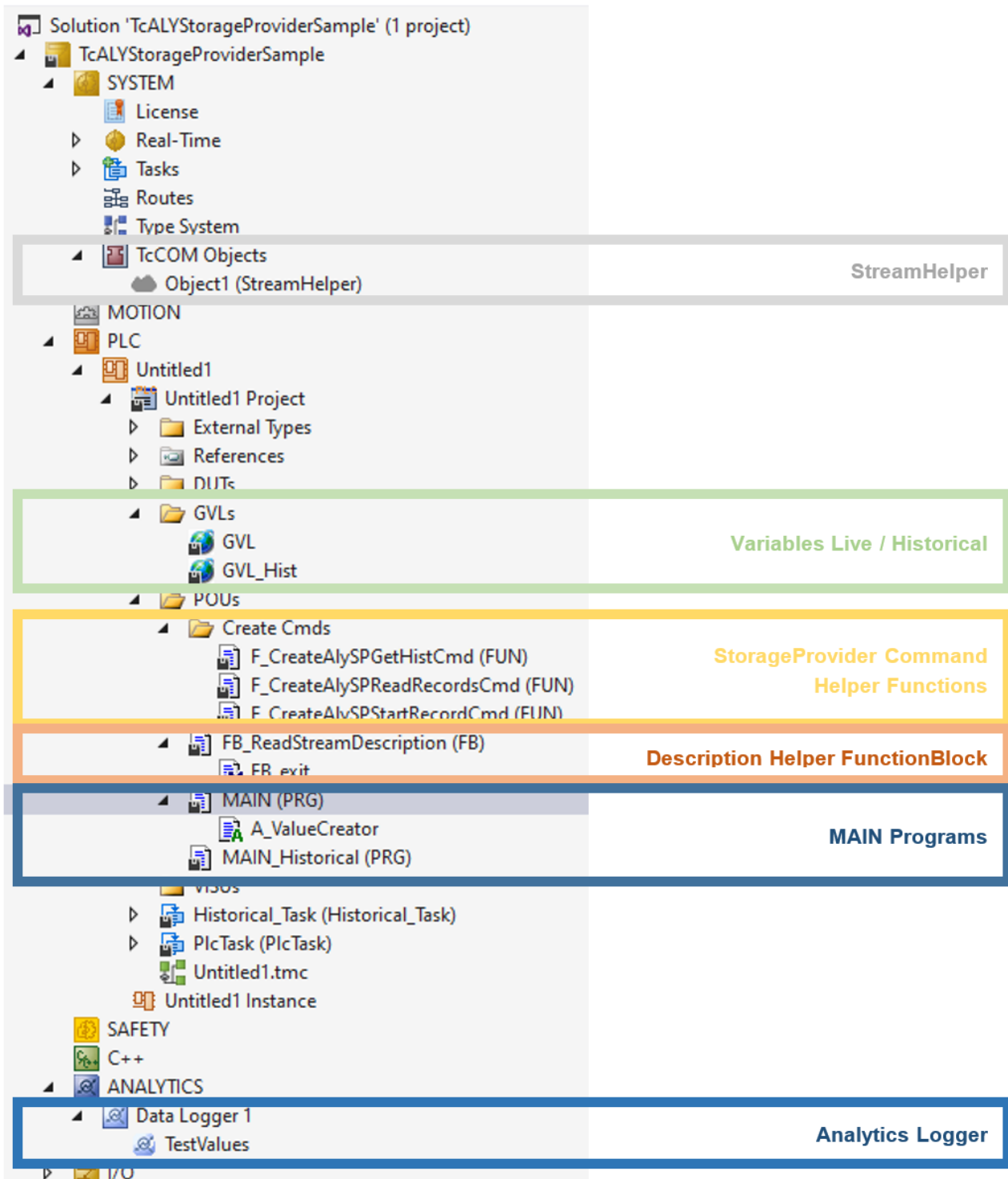
### Storage Provider Service GUID

GUID des individuell genutzten Storage Provider Services. Wenn Sie mit einem Storage Provider im Recorder Fenster verbunden sind, können Sie die Provider GUID den Properties entnehmen.



### Beispiel-Code-Architektur

Alle relevanten Stellen der Konfiguration und des Programm-Codes sind in dem nachfolgenden Bild markiert:



### Stream Helper

Für den Empfang der historischen Daten, die von dem Analytics Storage Provider per MQTT verschickt werden.

### Variable Live/Historisch

Die GVL ist für die Live-Daten und die GVL\_Hist ist für die historischen Daten.

### Storage Provider Command Helper Functions

Diese Helper Funktionen erzeugen die Kommandos für die Kommunikation mit dem Storage Provider Service im JSON Format.

**IsRecordingActive**

Hier wird gezeigt wie Sie z.B. die von dem Analytics Storage Provider gesendete Description mit Hilfe der JsonXML Bibliothek parsen und Informationen gewinnen, ob beispielsweise ein angegebener Record Alias Name aktuell aufgezeichnet wird.

**Description Helper Function Block**

Dieser Funktionsbaustein empfängt die Informationen des Description Topics u.a. mit den Informationen zur SystemID.

**MAIN Programs**

Das Main Programm ruft die Kommunikation zu dem Analytics Storage Provider auf. Das Main Historical Programm realisiert das Mapping der historischen Daten von dem Stream Helper in die GVL\_Hist.

**Analytics Logger**

Sendet die Variablen der GVL an einen MQTT Message Broker.

**Beispiel Starten**

Bevor das Beispiel gestartet werden kann, müssen Sie an drei unterschiedlichen Stellen den von Ihnen verwendeten MQTT Message Broker einstellen.

Analytics Stream Helper:

Object	Context	Parameter (Init)	Parameter (Online)	Interfaces
-	MQTT			
	Host Name	172.17.62.145	<input type="checkbox"/>	STRING(80) 0x020201...
	TcpPort	1883	<input type="checkbox"/>	UINT 0x020201...
	Topic Prefix	_AlySPTest/ResultValues	<input type="checkbox"/>	STRING(255) 0x020300...
	Client ID		<input type="checkbox"/>	STRING(80) 0x020201...
	Username		<input type="checkbox"/>	STRING(80) 0x020201...
	Password		<input type="checkbox"/>	STRING(80) 0x020201...
	Max Handles	100	<input type="checkbox"/>	UDINT 0x020300...
	Fifo Exponent	8	<input type="checkbox"/>	UDINT 0x020300...
	Sender System Id		<input type="checkbox"/>	STRING(80) 0x020300...
-	Tls			
	CaFile		<input type="checkbox"/>	STRING(255) 0x020201...
	CertFile		<input type="checkbox"/>	STRING(255) 0x020201...
	KeyFile		<input type="checkbox"/>	STRING(255) 0x020201...
	KeyPw		<input type="checkbox"/>	STRING(255) 0x020201...
	CrlFile		<input type="checkbox"/>	STRING(255) 0x020201...
	PskId		<input type="checkbox"/>	STRING(255) 0x020201...
	PskPw		<input type="checkbox"/>	STRING(255) 0x020201...
	Insecure	FALSE	<input type="checkbox"/>	BOOL 0x020201...
	Version		<input type="checkbox"/>	STRING(255) 0x020201...

Analytics Logger:

Context	Parameter (Int)	Data Area	TLS	Time Source		
	Name	Value	CS	Type	PTCID	
	Data Format	ANALYTICS_FORMAT_MQTT_BINARY	<input type="checkbox"/>	ANALYTICS_FORMAT	0x02020114	
	Data Compression	ANALYTICS_COMPRESSION_RL	<input type="checkbox"/>	ANALYTICS_COMPRESS...	0x02030027	
	Max. Compression Compare Width	ANALYTICS_COMP_WIDTH_8	<input type="checkbox"/>	ANALYTICS_COMPRESS...	0x02030030	
	MQTT Host Name	172.17.62.145	<input type="checkbox"/>	STRING(80)	0x02020113	
	MQTT Tcp Port	1883	<input type="checkbox"/>	UINT	0x02020103	
	MQTT Main Topic	AlySPTest	<input type="checkbox"/>	STRING(255)	0x02030008	
	MQTT Client ID		<input type="checkbox"/>	STRING(80)	0x02020101	
	MQTT User Name		<input type="checkbox"/>	STRING(255)	0x02020106	
	MQTT Password		<input type="checkbox"/>	STRING(80)	0x02020107	

## MAIN Programm:

```

MAIN  # X
12  VAR
13  nState: INT;
14  fbALY_StorageProvider: FB_ALY_StorageProvider := (stConfig := (sMainTopic := 'Beckhoff', sProviderGuid := STRING_TO_GUID('1f6e80ac-5250-4a2e-a105-c0c2d9a981fd'),
15  stConnSettings := (sHostName := '172.17.62.145', sUserID := '', sPassword := '', sClientId := 'PLCClient')));
16
17  fbReadStreamDescription: FB_ReadStreamDescription := (stConnSettings := (sHostName := '172.17.62.145', sUserID := '', sPassword := '', sClientId := 'PLCClient2'));
18
19  stStartCmd: T_ALY_SPRecordData_Cmd;
20  stStopCmd: T_ALY_SPRecordData_Cmd;
21  stReadRecordsCmd: T_ALY_SPReadStreamRecords_Cmd;
22  stGetHistDataCmd: T_ALY_SPGetHistorical_Cmd;
23
24  aRecordInfos: ARRAY [1..10] OF T_RecordTimestamps;
25  nRecs: DINT;
26  END_VAR

```

Anschließend müssen Sie die sProviderGUID für den FB\_ALY\_StorageProvider ändern. Zu finden ist diese wie oben in diesem Dokument beschrieben.

Für die Steuerung des Beispiels gehen Sie nun in das MAIN Programm. Mit dem Enum eCtrl können Sie die Aktion einstellen, die Sie gerne durchführen möchten. Es stehen zur Auswahl:

- StartRecord
- StopRecord
- ReadRecords
- GetHistorical

Mit einer steigenden Flanke an der Variablen bExecute wird die im Enum gewählte Aktion ausgeführt. Sollten Sie mehrere Records gemacht haben, können Sie dies in dem Array aRecordInfo einsehen. Mit dem Index ist es dann möglich, die unterschiedlichen Records auszuwählen. Die Zeitspannen werden ebenfalls angezeigt, diese könnten Sie theoretisch auch noch innerhalb der Zeitspanne anpassen. Dafür müssten Sie die Logik des Beispiels in der Helfer-Funktion F\_CreateAlySPGetHistCmd entsprechend anpassen.

Die oben im Dokument herausgesuchte Storage Provider Recorder GUID kann optional in der SPS in der Funktion F\_CreateAlySPStartRecordCmd gesetzt werden. Theoretisch kann es eine beliebige GUID sein, sie dient nur der Identifikation des Recorders – in diesem Fall der SPS. Stellt man jedoch die Recorder GUID aus dem eigenen Engineering ein, so wird der Record direkt im Analytics Storage Provider Recorder angezeigt. Wird eine andere GUID verwendet, muss ggf. erst die Filter Funktion im Recorder Fenster eingeschaltet werden, damit der Record aus der SPS angezeigt wird.

Download: [https://infosys.beckhoff.com/content/1031/tf3500\\_tc3\\_analytics\\_logger/Resources/11270100747.zip](https://infosys.beckhoff.com/content/1031/tf3500_tc3_analytics_logger/Resources/11270100747.zip)



## 9 Anhang

### 9.1 FAQ – Häufig gestellte Fragen und Antworten

In diesem Abschnitt werden häufig gestellte Fragen beantwortet, um Ihnen die Arbeit mit dem TwinCAT Analytics Storage Provider (ASP) zu erleichtern. Falls Sie weitere Fragen haben, wenden Sie sich bitte an unser Support-Team [support@beckhoff.com](mailto:support@beckhoff.com).

1. [Wie kann ich das Tabellenschema von MS SQL mit ASP verwalten? \[► 97\]](#)
2. [Kann ich den Storage Provider auf programmierbare Weise steuern? \[► 97\]](#)
3. [Ist es möglich, auch Ergebnisse von der Analytics Runtime zu speichern? \[► 97\]](#)
4. [Werden in TwinCAT Measurement Produkten Open source Softwarekomponenten verwendet? \[► 97\]](#)
5. [Von welchen Einflussfaktoren hängt der Datendurchsatz des Storage Providers ab? \[► 97\]](#)

#### Wie kann ich das Tabellenschema von MS SQL mit ASP verwalten?

Über das Tabellenschema müssen Sie sich keine Gedanken machen. Dies wird komplett vom Analytics Storage Provider übernommen. Sie müssen lediglich festlegen, auf welchem Datenbankserver die Daten gespeichert werden sollen. Wenn Sie Daten in Ihrer eigenen Tabellenstruktur sehen möchten, müssen Sie die Daten in eine TwinCAT Analytics Runtime streamen und vom TwinCAT Database Server in Ihrer Struktur schreiben lassen.

#### Kann ich den Storage Provider auf programmierbare Weise steuern?

Ja, durch die SPS-Schnittstelle für den TwinCAT Storage Provider. Sie können Aufzeichnungen starten/stoppen oder historische Daten (Rohdaten oder Ergebnisdaten) abrufen.

#### Ist es möglich, auch Ergebnisse von der Analytics Runtime zu speichern?

Ja, das ist möglich. Hierzu können Sie beim Generieren der Analytics Runtime aus dem Analytics Workbench-Konfigurator wählen, dass die Ergebnisse an einen MQTT-Message-Broker gesendet werden sollen. Dieser Datenstrom kann vom Storage Provider erfasst werden.

#### Werden in TwinCAT Measurement Produkten Open Source Softwarekomponenten verwendet?

Ja es werden verschiedene Open Source Komponenten verwendet.

Bitte sehen Sie dazu die Information auf der Seite [Third-party components \[► 98\]](#).

#### Von welchen Einflussfaktoren hängt der Datendurchsatz des Storage Providers ab?

Der Datendurchsatz hängt von vielen Einflussgrößen ab. In erster Linie von System- und Netzwerkressourcen. Eine Übersicht:

- Systemeigenschaften (CPU, RAM)
- Schreibgeschwindigkeit und Qualität des Speichermediums (SSD)
- Netzwerkeigenschaften
- Komplexität der Symbolik (Datentyp, Strukturen, Arrays, etc.)
- Modus der Historisierung (Gesamte Symbolik erlaubt höheren Durchsatz, eine Teilmenge kann je nach Größe aufwendiger sein)
- Komprimierungsgrad des Streams (je stärker die Komprimierung, desto höher die Systembelastung)
- Die Größe eines Samples
- Gesamtgröße eines Datenpakets (Anzahl Samples pro Paket)
- Die Anzahl paralleler Aufnahmen die der Storage Provider verwaltet

## 9.2 Third-party components

This software contains third-party components.

Please refer to the license file provided in the following folder for further information:

*...\TwinCAT\Functions\TwinCAT Measurement\Legal*



Mehr Informationen:  
**[www.beckhoff.com/tf3520](http://www.beckhoff.com/tf3520)**

Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Deutschland  
Telefon: +49 5246 9630  
[info@beckhoff.com](mailto:info@beckhoff.com)  
[www.beckhoff.com](http://www.beckhoff.com)

