zenon
by COPA-DATA

zenon Handbuch

HTML Web Engine

v.8.20

© 2020 Ing. Punzenberger COPA-DATA GmbH

Alle Rechte vorbehalten.

Die Weitergabe und Vervielfältigung dieses Dokuments ist - gleich in welcher Art und Weise - nur mit schriftlicher Genehmigung der Firma COPA-DATA gestattet. Technische Daten dienen nur der Produktbeschreibung und sind keine zugesicherten Eigenschaften im Rechtssinn. Änderungen - auch in technischer Hinsicht - vorbehalten.

Inhaltsverzeichnis

1 Willkommen bei der COPA-DATA Hilfe 5

2 HTML Web Engine 6

3 Erforderliche Komponenten und deren Definitionen 7

4 Grundlegender Systemaufbau der HTML Web Engine 9

5 Systemvoraussetzungen 12

6 Lizenzierung 14

7 Installation 15

7.1 IIS Publishing Dienst installieren 15

7.1.1 .NET-Registrierung am IIS unter Windows 7 21

8 Connectoren 22

8.1 SCADA Runtime Connector 23

8.2 Service Grid Connector 24

8.2.1 zenon Variablen 25

8.3 Reconnect 25

9 Client Authentifizierung für eine Verbindung der Runtime 26

10 Deployment der Web Engine 26

10.1 zenon Web Engine Deployment Tool 27

10.1.1 Allgemeine Einstellungen für die Web Engine 29

10.1.2 Einstellungen für den Runtime Connector 32

10.1.3 Einstellungen für den Service Grid Connector 35

10.1.4 Validierung der Einstellungen 37

10.1.5 Fortschritt und Ergebnis 39

11 Projektieren im Editor 39

11.1 Bild vom Typ HTML anlegen, anpassen und aufrufen 40

11.1.1 Optionen für das Öffnen externer Webseiten 43

11.2 Beispiel: Einfaches Startbild 44

11.3 Unterstützte Funktionalitäten für die HTML Visualisierung 45

11.3.1 AML und CEL 47

11.3.2 Bilder, Schablonen, Elemente und Symbole 53

11.3.3 Funktionen 64
11.3.4 Tonsignal oder Dauerton abspielen 67
11.3.5 Variablen 68

12 Projekt für Web kompilieren 70

13 Ablauf einer HTML Web Engine Sitzung 72

14 Systemdiagnose und Fehlersuche 73

1 Willkommen bei der COPA-DATA Hilfe

ZENON VIDEO-TUTORIALS

Praktische Beispiele für die Projektierung mit zenon finden Sie in unserem YouTube-Kanal (https://www.copadata.com/tutorial_menu). Die Tutorials sind nach Themen gruppiert und geben einen ersten Einblick in die Arbeit mit den unterschiedlichen zenon Modulen. Alle Tutorials stehen in englischer Sprache zur Verfügung.

ALLGEMEINE HILFE

Falls Sie in diesem Hilfekapitel Informationen vermissen oder Wünsche für Ergänzungen haben, wenden Sie sich per E-Mail an documentation@copadata.com.

PROJEKTUNTERSTÜTZUNG

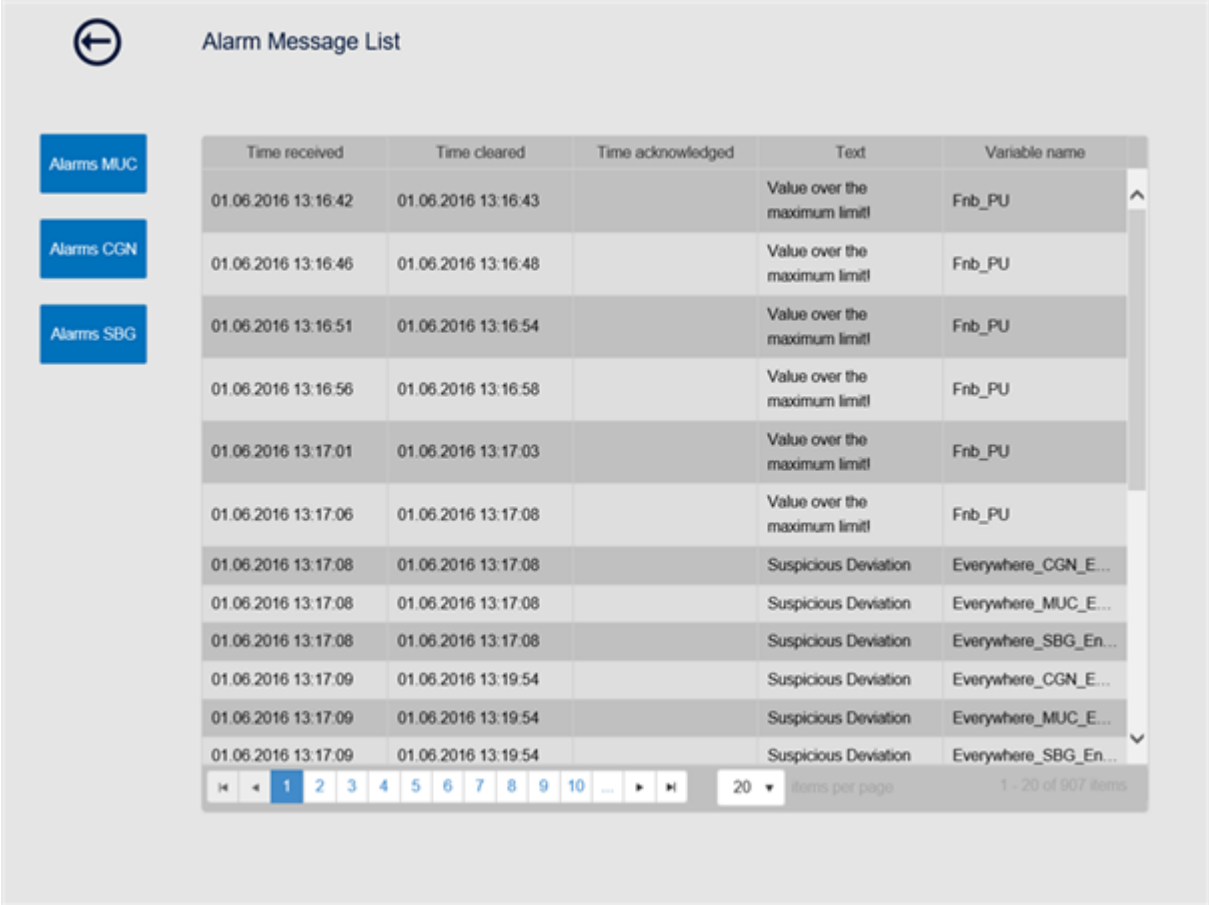
Unterstützung bei Fragen zu konkreten eigenen Projekten erhalten Sie vom Customer Service, den Sie per E-Mail an support@copadata.com erreichen.

LIZENZEN UND MODULE

Sollten Sie feststellen, dass Sie weitere Module oder Lizenzen benötigen, sind unsere Mitarbeiter unter sales@copadata.com gerne für Sie da.

2 HTML Web Engine

Die **HTML Web Engine** dient zur Bereitstellung von zenon Bildern als HTML5 Webseite. Am Visualisierungs-Endgerät wird per Webbrowser die Benutzerschnittstelle aufgerufen und dargestellt. Es sind dafür keine spezielle Software-Installation - auch keine Browser Plug-Ins - am Endgerät erforderlich. Prozessdaten für die Visualisierung werden von einer zenon Runtime bezogen.



FUNKTIONALITÄTEN DER HTML WEB ENGINE

Überblick über die Funktionalitäten der HTML Web Engine:

- Sitzungsbasierte Bereitstellung von HTML5 Visualisierungsinhalten an HTML Web Clients.
- Darstellung grundlegender Visualisierungsinhalte, die im zenon Editor erstellt wurden.
- Datenanzeige:
  - Variablenwerte
    Anzeigen werden in der HTML Web Engine dynamisch aktualisiert.
  - Chronologischen Ereignisliste (CEL)
    Einträge der CEL werden in der HTML Web Engine dynamisch aktualisiert.

- Erweiterter Trend (ETM)
  Die Darstellung im ETM wird in der HTML Web Engine nicht aktualisiert. Dies bedeutet, dass der ETM nur mit den beim Aufschalten vorhandenen Daten gezeichnet und dann nicht mehr aktualisiert wird.
- Meldungen der Alarmmeldeliste (AML)
  Einträge der AML werden in der HTML Web Engine dynamisch aktualisiert.

- Weiterleitung von Prozessinformationen, wie Variablenwerte, Alarm- oder Ereignis-Meldungen einer zenon Runtime an einen oder mehrere HTML Web Clients.
- Unterstützung aktiver Bedienhandlungen, wie Sollwert setzen.
- Mobiles, ortsunabhängiges Bedienen und Beobachten.
- Keine Installation und/oder Konfiguration am Endgerät, also am Client notwendig. Plattformunabhängige Darstellung in HTML5 Standard.
- Betrieb des HTML Web Servers auf einem separaten Rechner, wie zum Beispiel in einer DMZ möglich.
- Sichere Netzwerkkommunikation über HTTPS, basierend auf SSL-Zertifikaten.
- Schutz von sensiblen Visualisierungsbereichen oder Prozessen mittels Benutzerauthentifizierung und Unterstützung von Benutzerebenen.
- Abruf von Informationen aus dem **Service Grid**.

Die HTML Web Engine unterstützt die Authentifizierung eines Web Engine Clients mit erhöhter Sicherheit gegenüber der zenon Benutzer-Authentifizierung und gegenüber Active Directory. Die Anmeldung erfolgt durch Eingabe des Benutzernamens und Passwort.

REGIONALE EINSTELLUNGEN

Die HTML Web Engine unterstützt die meisten üblichen Sprachen, Zahlenformate und Datumsformate.

3 Erforderliche Komponenten und deren Definitionen

Komponente	Beschreibung
zenon Runtime	Die Prozessdaten für die HTML5 Visualisierung werden von einer zenon Runtime (Server oder Client) zur Verfügung gestellt.
HTML Web Engine (auf Seite 6)	Die HTML Web Engine dient zur Bereitstellung von Prozessbildern als HTML5 Webseite. Am Visualisierungs-Endgerät wird per Web Browser die Benutzerschnittstelle aufgerufen und dargestellt. Es

Komponente	Beschreibung
	sind dafür am Endgerät weder eine spezielle Software Installation noch ein Software Plug-In erforderlich. Prozessdaten für die Visualisierung werden von einer zenon Runtime bezogen. **Hinweis:** Die HTML Web Engine verarbeitet Prozessdaten zum Zweck der Visualisierung und Bedienung durch den HTML Web Client. Die Verwaltung der Prozessdaten erfolgt dabei ausschließlich durch die zenon Runtime.
IIS (auf Seite 15) **Publishing Dienst**	Diensteplattform von Microsoft für PCs und Server. Über sie können Dokumente und Dateien im Netzwerk zugänglich gemacht werden. Die HTML Web Engine nutzt den IIS als Laufzeitumgebung und zur Veröffentlichung von zenon Bildern. Als Kommunikationsprotokoll kommt HTTPS zum Einsatz. Mittels sogenanntem **Deployment** (auf Seite 26) wird die HTML Web Engine am IIS instanziert.
Web Browser	Web Browser sind spezielle Computerprogramme zur Darstellung von Dokumenten und Daten, speziell von Webseiten im World Wide Web.
zenon Web Engine Deployment Tool (auf Seite 26)	Stellt die HTML Web Engine als Web-Applikation im IIS bereit und ermöglicht deren Konfiguration. Es kann auch eine existierende HTML Web Engine Instanz aktualisiert oder gelöscht werden.
Web Engine Compiler	Generiert aus einem zenon Projekt heraus jene Daten, die die HTML Web Engine benötigt, um die HTML5-Inhalte für den zenon Web Client zur Verfügung zu stellen. Beim Übersetzungsvorgang dieser Projektdaten überprüft der HTML Web Engine Compiler die Projektinhalte und gibt Auskunft über nicht unterstützte Funktionen oder Eigenschaften. Als Resultat des Übersetzungsvorgangs entsteht eine Datei, die der Web Engine zur Verfügung gestellt wird.
SCADA Runtime Connector (auf Seite 23)	Dient als Kommunikationsschnittstelle zur zenon Runtime. Sie finden den **SCADA Runtime Connector** im Pfad: *%programfiles(x86)%\Common*

Komponente	Beschreibung
	Files\COPA-DATA\Connectors\zrsConnector.exe Wenn das Web Deployment Tool verwendet wird, muss der **SCADA Runtime Connector** ausgeführt werden, da dessen Status bei der Anwendung überprüft wird. Der Connector Container kann über das **Startup Tool** automatisch gestartet werden, wenn sich ein Benutzer am System anmeldet.
Service Grid Connector (auf Seite 24)	Der **Service Grid Connector** ermöglicht die Verbindung der HTML Web Engine mit der Service Grid Infrastruktur. Damit kann die HTML Web Engine auch außerhalb eines SCADA-Netzwerks betrieben werden.

4 Grundlegender Systemaufbau der HTML Web Engine

Die HTML Web Engine stellt als Web Applikation eine HTML5 Webseite zur Verfügung.

Achtung

Empfehlungen:

- Betreiben Sie die gesamte Systemkonfiguration immer in einem vertrauenswürdigen Netzwerkbereich.
- Publizieren Sie HTML5 Website nie direkt im Internet.

Im Verlauf einer Sitzung wird zwischen zwei unterschiedlichen Verbindungsstufen unterschieden:

1. Darstellung der Visualisierungsseiten ohne Prozessdaten
2. Darstellung der Visualisierungsseiten und Anzeige von Prozessdaten

DARSTELLUNG DER VISUALISIERUNGSSEITEN OHNE PROZESSDATEN

Der zenon Web Client verbindet sich zur HTML Web Engine durch Aufruf des URL (Uniform Ressource Locator) für die HTML5 Webseite. Nach erfolgreichem Aufbau der Sitzung kann das Projekt ohne Zugriff auf Prozessdaten der zenon Runtime visualisiert werden.

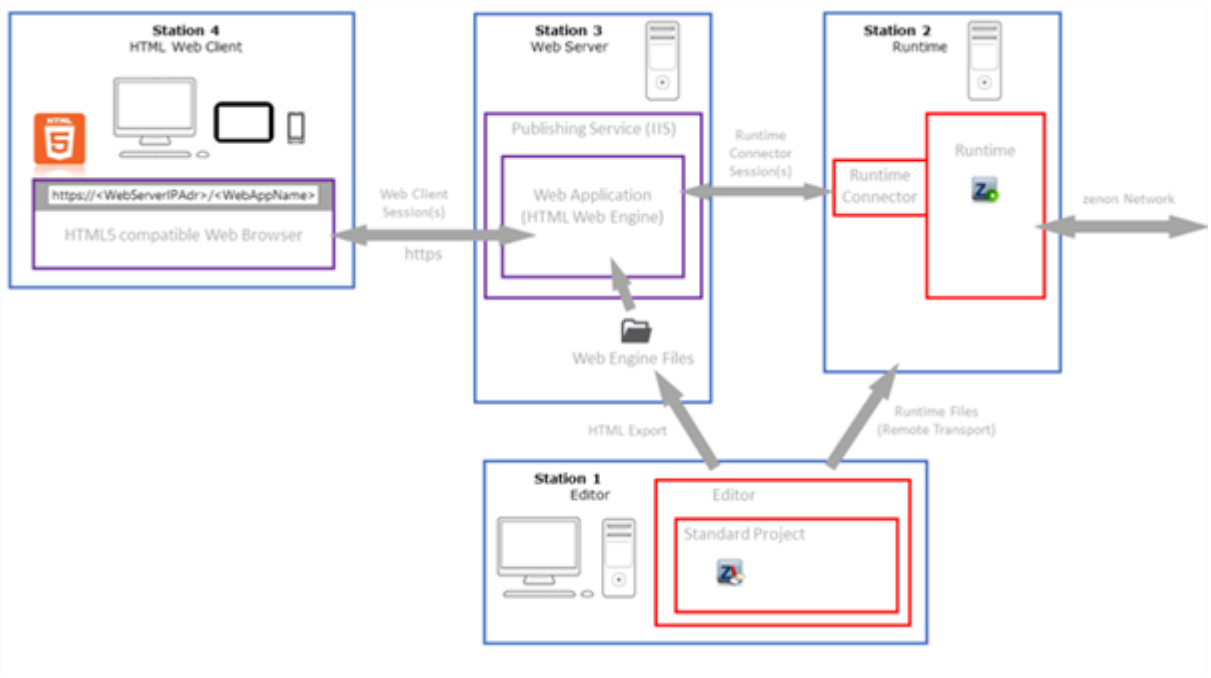
DARSTELLUNG DER VISUALISIERUNGSSEITEN UND ANZEIGE VON PROZESSDATEN

Die HTML Web Engine verbindet sich über den **SCADA Runtime Connector** (auf Seite 23) oder den **Service Grid Connector** (auf Seite 24) zur zenon Runtime. Diese Verbindung wird nur zugelassen, wenn eine Benutzerauthentifizierung auf Basis von Benutzername und Passwort erfolgreich durchgeführt wurde. Die Authentifizierung erfolgt mit externer Authentifizierung an der Benutzerverwaltung der zenon Runtime. Die Übergabe der Benutzerinformation kann entweder manuell durch den zenon Web Client Bediener oder automatisch durch die Web Engine erfolgen.

Zur Projektierung der HTML5 Visualisierung sind keine speziellen Werkzeuge erforderlich. Die Erstellung der Bilder und Funktionen erfolgt standardmäßig im zenon Editor.
Eine Auflistung der unterstützten Elemente, Eigenschaften und Funktionen finden Sie im Kapitel **Unterstützte Funktionalitäten für die HTML5 Visualisierung** (auf Seite 45).

ZUGRIFF VON EINEM ENDGERÄT AUF DIE HTML5 VISUALISIERUNG

So greifen Sie mit einem HTML5-fähigen Web Browser von einem Visualisierungs-Endgerät aus auf die HTML5 Visualisierung zu:



Hinweis: Die Aufteilung der Komponenten dient nur der einfachen Darstellung. Die gesamte hier dargestellte Konfiguration kann in beliebiger Verteilung der Komponenten in einem Netzwerk oder auch auf einem einzelnen Rechner betrieben werden.

Ablauf:

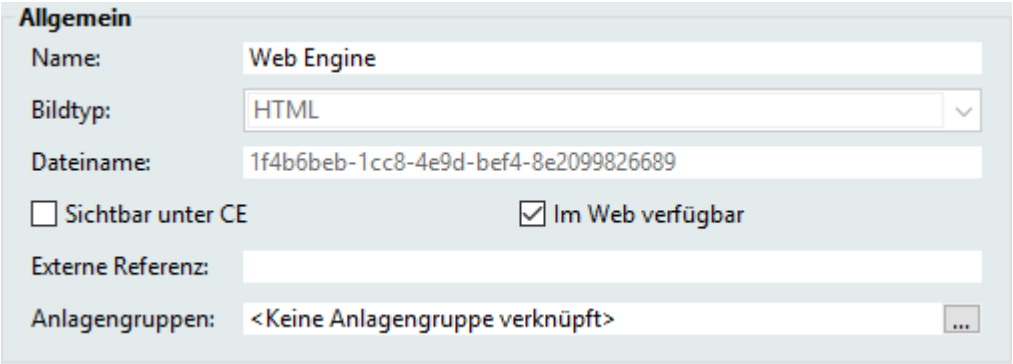
1. Der Bediener auf Station 4 verbindet sich mit einem Standard Web Browser durch Eingabe der Webseiten-URL zum Web Server auf Station 3.
   Dadurch erhält er von der Web Engine die dort vorliegenden Visualisierungsseiten.
2. Prozessdaten werden in der HTML5 Visualisierung erst nach einer erfolgreichen Benutzerauthentifizierung angezeigt.
   Nach erfolgreicher Prüfung von Benutzername und Passwort wird die Verbindung zur zenon Runtime (im Beispiel auf Station 2) aufgebaut.
3. Die Schnittstelle zwischen zenon Runtime und Web Engine bildet der **SCADA Runtime Connector** oder der **Service Grid Connector**.
4. Die Projektierung der HTML5 Visualisierung wird aus einem zenon Editor-Projekt (im Beispiel auf Station 1) abgeleitet.
   Die Projektstände auf Station 2 (zenon Runtime) und Station 3 (Web Engine) sollten dabei identisch sein.
   Mit dem **Web Engine Compiler** werden die Projektdaten für die Verwendung durch die HTML Web Engine übersetzt. Die notwendigen Schritte zur Installation der HTML Web Engine auf Station 3 und das Deployment am Internet Information Server werden im Kapitel Deployment der Web Engine (auf Seite 26) beschrieben.

   Die Datei aus dem **Web Engine Compiler** wird in einem frei definierbaren Ordner abgelegt. Hier wird sie von der HTML Web Engine ausgelesen.

Hinweise: Stellen Sie für die Anbindung einer HTML5-Visualisierung eine dedizierte zenon Runtime zur Verfügung, um die reibungslose Zusammenarbeit mit prozessrelevanten Abläufen zu garantieren. Führen Sie nach Änderungen im zenon Projekt den Kompiliervorgang noch einmal durch.

BILDER FREIGEBEN

Mittels der HTML Web Engine können bestimmte Bilder aus dem aktiven Projekt in einem Web Browser aufgerufen werden. Welche Bilder für die Darstellung im Web Browser freigegeben werden, bestimmen Sie für jedes Bild über die Eigenschaft **Im Web verfügbar**.



Alle Bilder, für die diese Eigenschaft aktiviert wurde, werden mit dem **Web Engine Compiler** für die Web Engine kompiliert und stehen für die Bereitstellung im Web Browser auf dem Client zur

Verfügung. Bilder, für die diese Eigenschaft nicht aktiviert wurde, können für die zenon Runtime Visualisierung verwendet werden, sind jedoch nicht im Web Browser verfügbar.

Folgende Bildtypen können für die Visualisierung im Web genutzt werden:

- *AML*
- *CEL*
- *ETM*
- *Login*
- *HTML*
- *Standard*

5 Systemvoraussetzungen

WEB SERVER

Die HTML Web Engine unterstützt die folgenden Betriebssysteme.

Unterstützte Desktop Betriebssysteme und erforderliche Service Packs:

Betriebssystem	Service Pack
Windows 7 (Professional, Enterprise und Ultimate Version, jeweils x86 und x64 Versionen)	SP 1
Windows Embedded Standard 7 (vorausgesetzt, alle notwendigen Betriebssystemkomponenten sind vorhanden)	SP 1
Windows 8 und 8.1 (Standard, Professional, Enterprise, jeweils x86 und x64 Versionen)	SP 0
Windows Embedded 8 Standard (vorausgesetzt, alle notwendigen Betriebssystemkomponenten sind vorhanden)	SP 0
Windows 10 (Home, Pro, Enterprise, Education, Pro Education, Enterprise LTSB/LTSC, IoT Enterprise, Pro for Workstations)	SP 0

Unterstützte Server Betriebssysteme und erforderliche Service Packs:

Server Betriebssystem	Service Pack
Windows Server 2008 R2 (Alle Editionen, ausgenommen Core)	SP 1
Windows Server 2012 und 2012 R2 (Alle Editionen, ausgenommen Core)	SP 0
Windows Server 2016 (Alle Editionen, ausgenommen Core)	SP 0

Hinweis: Für den Betrieb der HTML Web Engine ist das .NET Framework 4.5 oder höher erforderlich.

Info

Beachten Sie die Limitierung der Anzahl gleichzeitiger Client Verbindungen durch den Microsoft IIS.

Hinweis: Bei Verwendung von Windows Server Betriebssystemen wird die Anzahl der gleichzeitigen Client Verbindungen durch den Microsoft IIS nicht limitiert. Bei Verwendung von Windows Desktop Betriebssystemen variiert diese Anzahl abhängig von der Version des installierten Betriebssystem.

WEB CLIENT

Für den HTML Web Client ist keine spezielle Installation erforderlich. Er funktioniert in der Regel mit jedem Web Browser, der folgenden Technologien unterstützt:

- HTML5
- HTML5 Canvas
- JavaScript (ECMAScript 5.1)

Die Verarbeitung der HTML5 Inhalte erfolgt im Web Browser unabhängig vom Betriebssystem. Es wird empfohlen, eine aktuelle Version eines der folgenden Web Browser zu verwenden:

- Windows Internet Explorer ab Version 11
- Microsoft Edge
- Mozilla Firefox
- Apple Safari
  Ausnahme: Unterstützt nicht das Abspielen einer Audiodatei.
- Google Chrome

Hinweis: Javascript muss im Web Browser aktiviert sein.

CLIENT-SERVER VERBINDUNG

Für die Verbindung zwischen Web Server und HTML Client muss eine ausreichende Datenrate zur Verfügung stehen. Bei einer zu geringen Datenrate werden entsprechende Meldungen am HTML Web Client angezeigt. Eine zu geringe Datenrate kann dazu führen, dass eine Sitzung nicht zustande kommt oder abgebrochen werden muss. Hinweise zum Verhalten bei Verbindungsabbruch finden Sie im Kapitel **Reconnect** (auf Seite 25).

Hinweis: Bei Verwendung von Windows Server Betriebssystemen wird die Anzahl der gleichzeitigen Client Verbindungen durch den Microsoft IIS nicht limitiert. Bei Verwendung von Windows Desktop Betriebssysteme hängt die mögliche Anzahl gleichzeitiger Verbindungen von der Version des installierten Betriebssystems ab.

6 Lizenzierung

Die HTML Web Engine muss pro Instanz lizenziert werden. Die **Lizenzierung** erfolgt über die **Lizenzverwaltung**. Bei fehlender Lizenz wird die HTML Web Engine in einem zeitlich begrenzten Demo-Modus gestartet.

Folgende Ausbauten stehen für die HTML Web Engine zur Verfügung:

- *Standard*: Zugriff auf die Visualisierung nur lesend. Der **HTML Web Client** kann als Betrachter verwendet werden.
- *Pro*: Vollzugriff auf die Visualisierung, lesend und schreibend.

> **Info**
>
> Beachten Sie die Limitierung der Anzahl gleichzeitiger Client Verbindungen durch den Microsoft IIS.
>
> **Hinweis:** Bei Verwendung von Windows Server Betriebssystemen wird die Anzahl der gleichzeitigen Client Verbindungen durch den Microsoft IIS nicht limitiert. Bei Verwendung von Windows Desktop Betriebssystemen variiert diese Anzahl abhängig von der Version des installierten Betriebssystem.

Unterschiede HTML Web Engine zu zenon Web Server:

- Die Lizenzüberprüfung erfolgt auf Basis von Instanzen. Jedes Projekt entspricht einer Instanz.
- Wird eine Lizenz ungültig, dann wird erst eine weitere Lizenz gesucht. Wird keine gefunden, kann über die Schaltfläche zur Lizenzsuche nach einer weiteren gültigen Lizenz gesucht werden.

- Laufen Demo-Lizenzen ab, muss der Service neu gestartet werden, um den Demo-Modus nutzen zu können. Der Demo-Modus steht nur zur Verfügung, wenn eine Demo-Lizenz hinterlegt ist.

7 Installation

Für den Betrieb der HTML Web Engine benötigen Sie folgende Installationen:

Web Server:

- IIS Publishing Dienst.

  Richten Sie den Publishing Dienst gemäß den Anleitungen aus Kapitel IIS Publishing Dienst installieren (auf Seite 15) ein.

- zenon Web Server.

  Installieren Sie vom Installationsmedium den zenon Web Server.

  Microsoft Web Deploy wird beim Setup automatisch mitinstalliert.

Projektierung und Laufzeit-Anwendung:

- zenon (Editor und Runtime).

Hinweis: Für den HTML Web Client ist keine spezielle Installation erforderlich.

7.1 IIS Publishing Dienst installieren

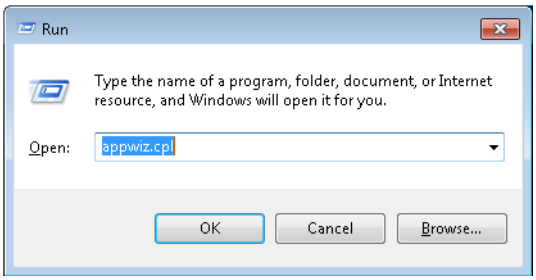
Internet Information Services, abgekürzt IIS, dienen zur Veröffentlichung von Dokumenten, wie zum Beispiel HTML-Seiten, über das HTTP-Protokoll. In den im Kapitel Systemvoraussetzungen (auf Seite 12) genannten Betriebssystemen ist der IIS Publishing Dienst bereits in der Standard-Installation enthalten. Dieser muss nur noch über die Windows Features aktiviert werden.

IIS 7, WINDOWS 7

Um den IIS Publishing Dienst zu aktivieren:

1. Drücken Sie Tastenkombination **Windows+R**.

Der Dialog zur Eingabe eines Befehl für die Kommandozeile wird geöffnet.

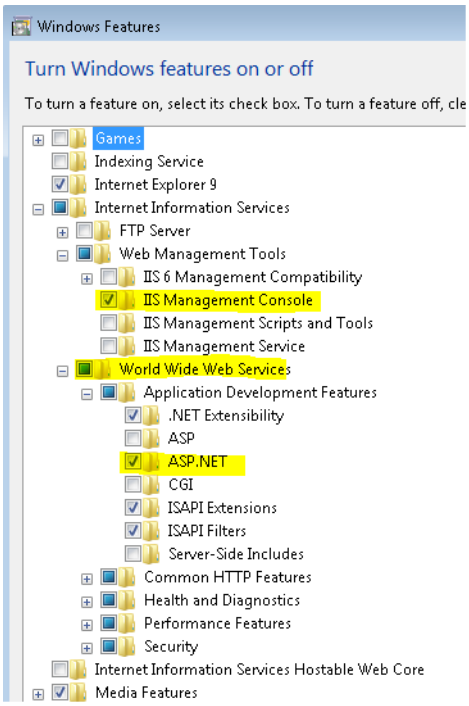


2. Geben Sie in das Eingabefeld *appwiz.cpl* ein.

   Klicken Sie auf **OK**.
   Ein neues Fenster der Systemsteuerung zur Konfiguration von Programmen und **Windows-Features** wird geöffnet.

3. Klicken Sie in diesem Fenster auf **Windows-Features aktivieren oder deaktivieren**.

   Das Fenster zur Auswahl von Features des Betriebssystems wird geöffnet.



4. Expandieren Sie in diesem Fenster den Knoten **Internetinformationsdienste**.
5. Aktivieren Sie *WWW-Dienste*.
   Damit werden in allen Unterordnern der Eigenschaft die Default-Einstellungen gesetzt.
6. Expandieren Sie den Knoten **Anwendungsentwicklungsfeatures**.
7. Aktivieren Sie **ASP.NET**.
8. Expandieren Sie den Knoten **Webverwaltungstools**.
9. Aktivieren Sie dort **IIS Verwaltungskonsole**.

10. Klicken Sie auf **OK**.

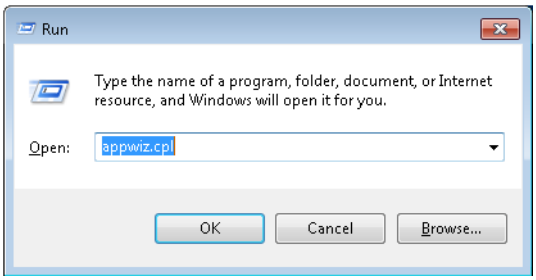
Hinweis: Bei nachträglicher Installation oder Upgrade des NET-Frameworks unter Windows 7, muss diese Software manuell in den IIS Publishing Dienst registriert werden. Mehr Details dazu siehe Kapitel **.NET-Registrierung am IIS unter Windows 7** (auf Seite 21).

IIS 8, WINDOWS 8/8.1

Um den IIS Publishing Dienst zu aktivieren:

1. Drücken Sie Tastenkombination **Windows+R**.

   Der Dialog zur Eingabe eines Befehl für die Kommandozeile wird geöffnet.

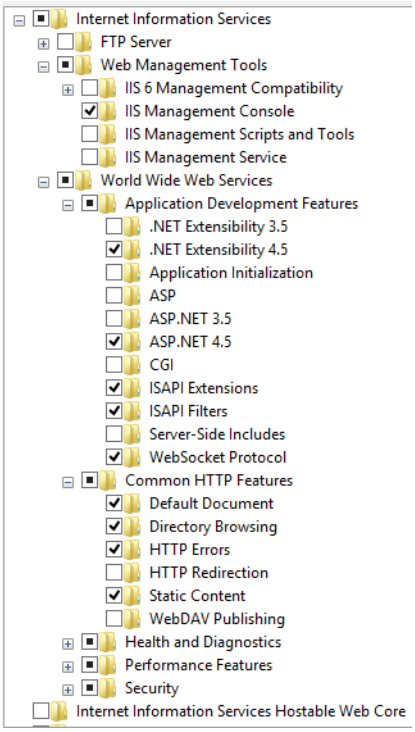


2. Geben Sie in das Eingabefeld *appwiz.cpl* ein.

   Klicken Sie auf **OK**.
   Ein neues Fenster der Systemsteuerung zur Konfiguration von Programmen und **Windows-Features** wird geöffnet.

3. Klicken Sie in diesem Fenster auf **Windows-Features aktivieren oder deaktivieren**.

Das Fenster zur Auswahl von Features des Betriebssystems wird geöffnet.



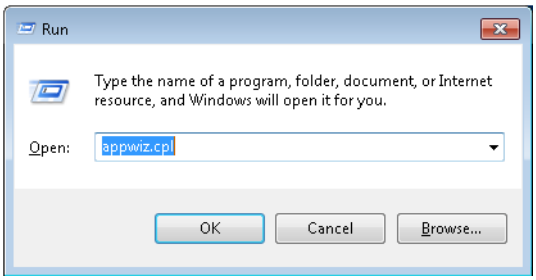
4. Expandieren Sie in diesem Fenster den Knoten **Internetinformationsdienste**.
5. Aktivieren Sie dort alle *WWW-Dienste*.
6. Expandieren Sie den Knoten **Anwendungsentwicklungsfeatures**.
7. Aktivieren Sie *ASP.NET 4.5*
8. Expandieren Sie den Knoten **Allgemeine HTTP-Features**
9. Aktivieren Sie dort *Statischer Inhalt*.
10. Expandieren Sie den Knoten **Webverwaltungstools**.
11. Aktivieren Sie dort optional *IIS Verwaltungskonsole*.
12. Expandieren Sie den Knoten **Anwendungsentwicklungsfeatures**.
13. Aktivieren Sie dort *WebSocket-Protokoll*.
14. Klicken Sie auf **Ok**.

WINDOWS 10

Um den IIS Publishing Dienst zu aktivieren:

1. Drücken Sie Tastenkombination **Windows+R**.

Der Dialog zur Eingabe eines Befehl für die Kommandozeile wird geöffnet.

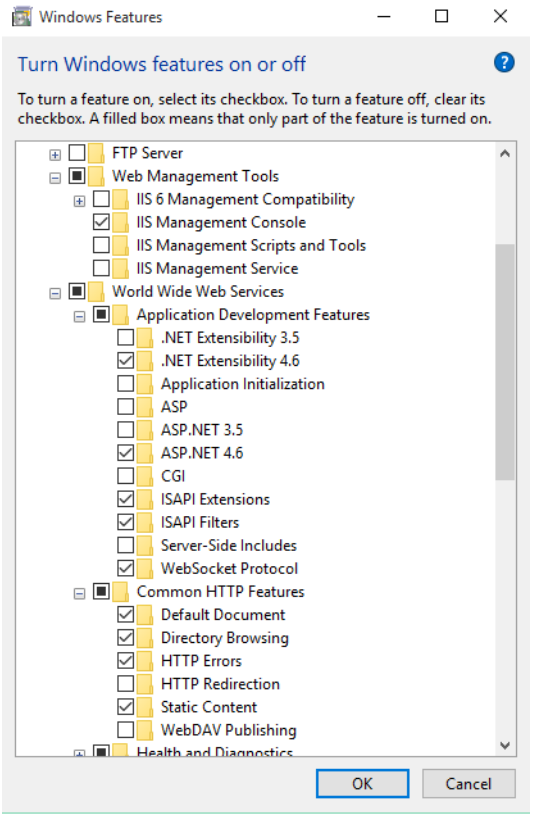


2. Geben Sie in das Eingabefeld *appwiz.cpl* ein.

   Klicken Sie auf **OK**.
   Ein neues Fenster der Systemsteuerung zur Konfiguration von Programmen und **Windows-Features** wird geöffnet.

3. Klicken Sie in diesem Fenster auf **Windows-Features aktivieren oder deaktivieren**.

   Das Fenster zur Auswahl von Features des Betriebssystems wird geöffnet.



4. Expandieren Sie in diesem Fenster den Knoten **Internetinformationsdienste**.
5. Aktivieren Sie dort alle *WWW-Dienste*.
6. Expandieren Sie den Knoten **Anwendungsentwicklungsfeatures**.
7. Aktivieren Sie *ASP.NET 4.7*
8. Expandieren Sie den Knoten **Allgemeine HTTP-Features**

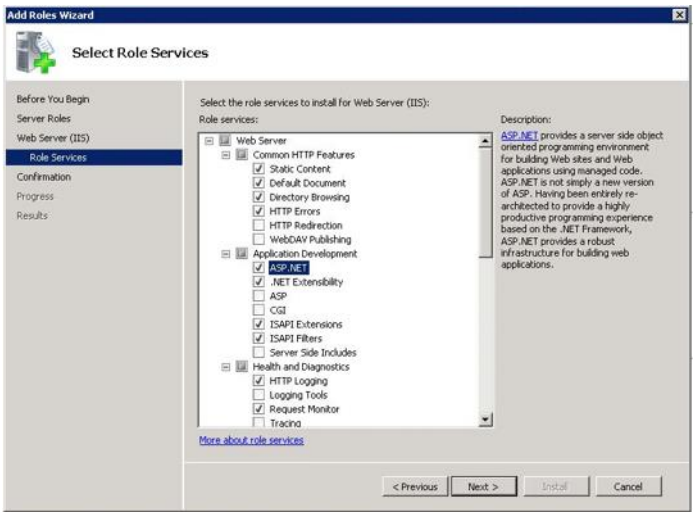
9. Aktivieren Sie dort *Statischer Inhalt*.
10. Expandieren Sie den Knoten **Webverwaltungstools**.
11. Aktivieren Sie dort optional *IIS Verwaltungskonsole*.
12. Expandieren Sie den Knoten **Anwendungsentwicklungsfeatures**.
13. Aktivieren Sie dort *WebSocket-Protokoll*.
14. Klicken Sie auf **Ok**.

WINDOWS SERVER 2008 R2

Folgen Sie den Anweisungen von Microsoft:
https://technet.microsoft.com/en-us/library/cc771209.aspx

1. Öffnen Sie den Assistenten **Rollen hinzufügen**.
2. Aktivieren Sie die *Web Server (IIS)*-Rolle

   Der Assistent **Rollen hinzufügen** öffnet sich.



3. Klicken Sie auf **Rollendienste**.
4. Expandieren Sie den Knoten **Anwendungsentwicklungfeatures**.
5. Aktivieren Sie die folgenden Rollendienste:

- ASP.NET
- .NET Erweiterbarkeit
- ISAPI-Erweiterungen
- ISAPI-Filter

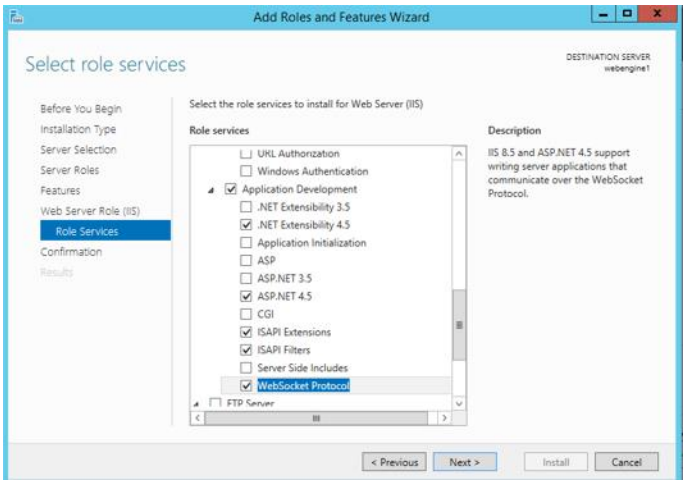
Achtung: Deaktivieren Sie keine Rollendienste, die schon von Microsoft vorausgewählt sind.

Hinweis: Es wird empfohlen Windows Server 2012 zu verwenden, da der Windows Server 2008 R2 keine WebSocket-Protokolle unterstützt.

WINDOWS SERVER 2012 (R2)

Folgen Sie den Anweisungen von Microsoft:
https://technet.microsoft.com/en-us/library/hh831475.aspx

1. Öffnen Sie den **Assistenten zum Hinzufügen von Rollen und Features** Wizard.



2. Expandieren Sie den Knoten **Anwendungsentwicklung**.
3. Aktivieren Sie die folgenden Rollendienste:
   - NET Erweiterbarkeit 4.5
   - ASP.NET 4.5
   - ISAPI Erweiterung
   - ISAPI-Filter
   - WebSocket Protokoll

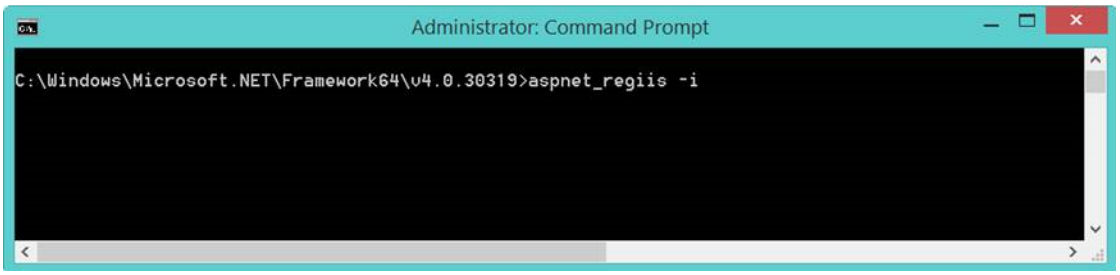
7.1.1 .NET-Registrierung am IIS unter Windows 7

Im Falle einer nachträglichen Installation oder Upgrade des .NET Frameworks unter Windows 7 muss eine Registrierung am IIS Publishing Dienst erfolgen.

Dazu:

1. Öffnen Sie die Windows Eingabeaufforderung mit der Option *Als Administrator*.
2. Wechseln Sie in den Microsoft.NET Installationsordner mit der höchsten Version.

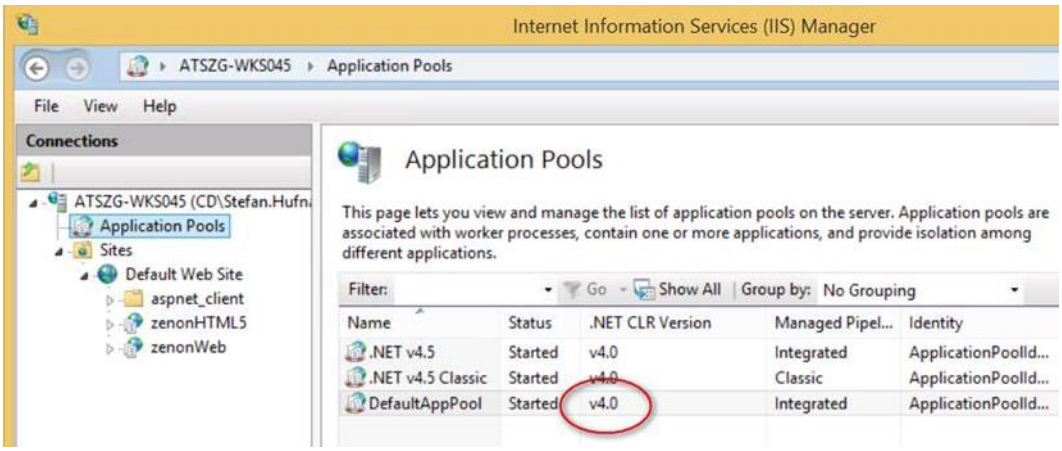
3. Geben Sie den Befehl **aspnet_regiis -i** ein.



Nach erfolgreicher Registrierung steht die aktuelle Microsoft .NET Framework Version für die Verwendung mit IIS bereit.

4. Stellen Sie sicher, dass der Application Pool am IIS, in dem die HTML Web Engine betrieben wird, die aktuelle .NET Version verwendet.

   Dies kann mit dem Internet Information Services Manager überprüft und eingestellt werden:



8 Connectoren

Damit die HTML Web Engine Daten von der zenon Runtime abfragen und darstellen kann, wird die Verbindung über einen Connector benötigt.

Zur Verfügung stehen:

- **SCADA Runtime Connector** (auf Seite 23)
- **Service Grid Connector** (auf Seite 24)

Die Benutzeranmeldung ist für beide Connectoren mit den Anmeldedaten des zenon Benutzers möglich.

8.1 SCADA Runtime Connector

Der **SCADA Runtime Connector** wird bei der Installation der zenon Runtime mitinstalliert. Die Interaktion beginnt grundsätzlich mit der Benutzerauthentifizierung für den ersten HTML Web Client.

VORAUSSETZUNGEN

Der **SCADA Runtime Connector** kommuniziert über das COM Interface.
In der zenon6.ini müssen eingetragen sein:

[VBA]

EIN=*1*

EVENT=*1*

STARTEN

Der SCADA Runtime Connector kann manuell oder automatisch gestartet werden:

- Manuell:
  - Starten Sie die Anwendung im **Startup Tool** unter **Tools**. Wählen Sie unter **Available applications (current folder)** den Eintrag **SCADA Runtime Connector** aus. Bestätigen Sie die Auswahl mit einem Klick auf **Start**.
  - Starten Sie die Anwendung direkt aus dem Ordner *%programfiles(x86)%\Common Files\COPA-DATA\Connectors\zrsConnector.exe*
- Automatisch:
  - Aktivieren Sie den Autostart für den Connector Container im **Startup Tool**.

Für jede HTML Web Engine Sitzung wird eine eigene **SCADA Runtime Connector** Sitzung aufgebaut.
In dieser Sitzung werden übertragen:

- Variablen:
  - Bild-Variablen, die für die aktuelle Bilddarstellung am HTML Client benötigt werden: Zum Beispiel Variablen für die Anzeige von Variablenwerten oder Elementdynamiken. Variablen können für eine Sitzung an- und abgemeldet werden. Nach erfolgreicher Benutzerauthentifizierung für einen HTML Web Client meldet die HTML Web Engine eine Liste von Variablen zur Kommunikation und spontanen Aktualisierung über den SCADA Runtime Connector an. Somit kann die HTML Web Engine Werteänderungen an die gerade verbundenen Web Clients weiterverteilen.
  - **Permanent überwachte Variablen**, die beständig angemelde (auf Seite 68)t bleiben und für die Ausführung einer Funktion bei Grenzwertverletzung benötigt werden.
- Sollwert setzen über den SCADA Runtime Connector:
  Zur Erhöhung der Sicherheit wird vor einer Wertänderung an der zenon Runtime eine

explizite Prüfung der Authentizität auf Basis der Benutzerdaten des **HTML Web Client** durchgeführt. Eine Sperre oder Entfernung von Benutzern auf Seiten der zenon Runtime wird somit für das Setzen von Sollwerten unmittelbar wirksam.

VERSCHLÜSSELTE KOMMUNIKATION

Die TCP-Verbindung zwischen **SCADA Runtime Connector** (**zrsConnector.exe**) und **SCADA Runtime Connector Client** (**zrsConnCli.dll**) kann mit AES verschlüsselt werden.

Um die verschlüsselte Kommunikation zu nutzen, vergeben Sie für Runtime und Client ein Verschlüsselungspasswort.
Dazu:

1. Tragen Sie im zenon Startup Tool in der Registerkarte **Network configuration** das Passwort ein.
   Dieses wird auch in der zenon6.ini verschlüsselt hinterlegt.
   Abschnitt: **[ZRSCONNECTOR]**
   Eintrag: **ENCRYPTION_PWD**
2. Setzen Sie im **Deployment Tool** der **HTML Web Engine** im Fenster **Security options** das gleiche Passwort in der Registerkarte **Security options** (auf Seite 32). Dieses wird verschlüsselt in der **web.config** in der Einstellung **Encryption_Password** abgelegt.

Ist bei der Validierung der **SCADA Runtime Connector** am eingestellten Zielrechner erreichbar, aber das Verschlüsselungspasswort stimmt nicht überein, dann läuft der Connector-Test in ein Timeout. Sie erhalten einen Hinweis auf ein eventuell falsch gesetztes Passwort.

8.2 Service Grid Connector

Der **Service Grid Connector** ermöglicht die Verbindung der HTML Web Engine mit der Service Grid Infrastruktur. Damit kann die HTML Web Engine auch außerhalb eines SCADA-Netzwerks betrieben werden.

Hinweis: Die HTML Web Engine erkennt über den **Service Grid Connector**, wenn die Verbindung zur zenon Runtime getrennt wurde. Bei Verbindungsabbruch werden die aktiven HTML Web Engine Benutzer automatisch abgemeldet. Im Fall einer konfigurierten Redundanz (Server/Standby) wird ein Wartedialog angezeigt und die HTML Web Engine versucht sich mit dem Standby-Server zu verbinden, der nun zum prozessführenden Server wird. Falls dies nicht gelingt, erscheint eine Fehlermeldung.

Voraussetzung: Die zenon Runtime, deren Visualisierung über die HTML Web Engine gewünscht ist, muss in die Service Grid Infrastruktur eingebunden sein.

Details zum Service Grid finden Sie im Handbuch Service Grid.

8.2.1 zenon Variablen

Um zenon Variablen im Service Grid zu nutzen, müssen diese dafür konfiguriert werden.
Hinweis: Es eigenen sich nur Variablen mit **einfachem Datentyp**. Strukturvariablen und Stammvariablen von Arrays stehen nicht zur Verfügung.

Tipp Alarme: Werden nicht alle erwarteten Alarme angezeigt, überprüfen Sie die Konfiguration der Variablen für Service Grid.

Um Variablen zu konfigurieren:

1. Wählen Sie die gewünschte Variable aus.
2. Öffnen Sie in den Eigenschaften die Gruppe **Externe Einstellungen**.
3. Wechseln Sie in die Untergruppe **Service Grid Einstellungen**.
4. Konfigurieren Sie die Variable für die Verwendung im Service Grid.

Konfigurierbare Eigenschaften:

- **Zugriffsberechtigung**: Konfiguration, welche Zugriffsberechtigungen im Service Grid auf eine Variable bestehen.
  Auswahl aus Dropdownliste:
  - *Keine*: Im Service Grid steht diese Variable nicht zur Verfügung.
  - *Lesend*: Im Service Grid kann lesend auf diese Variable zugegriffen werden.
  - *Lesend und schreibend*: Im Service Grid kann diese Variable gelesen und beschrieben werden.

  Default: *Keine*

Details zu Variablen in zenon lesen Sie im zenon Handbuch **Variablen**.

8.3 Reconnect

Wenn die Verbindung zwischen Web Engine und Runtime oder Service Grid verloren geht, wird versucht, die Verbindung wieder herzustellen.

Nach erfolgreicher Neuverbindung wird, falls konfiguriert, auch das Login automatisch ausgeführt. Wurde bei der Trennung auch die gerade aktive Benutzersitzung getrennt, wird versucht, auch diese wiederherzustellen.

Die automatische Wiederverbindung wird in folgenden Situationen durchgeführt:

- Unterbrechung der Verbindung zwischen Web Engine Frontend und Backend.
  Zum Beispiel Verbindungsabbruch bei mobile Verbindung oder Neustart des IIS.
- Verbindungsverlust zwischen Web Engine Backend und Connector/Data Hub.
- Verbindungsverlust durch Nachladen der zenon Runtime.

Hinweis Service Grid Connector: Bei Netzwerkprojekten für die Web Engine müssen die Adressen der Server mit ihrem vollqualifizierten Domainnamen (FQDN) eingetragen werden. Denn die IDs der Clients werden vom FQDN des Rechners abgeleitet. Wenn sich die konfigurierten Netzwerknamen und die FQDNs der Rechner unterscheiden, kann der Service Grid Connector den Zustand der zenon Runtime nicht korrekt erkennen.

9 Client Authentifizierung für eine Verbindung der Runtime

Laufzeit-Daten der zenon Runtime, wie etwa Variablenwerte zur Anzeige oder für Anzeige-Dynamiken werden nur zur Verfügung gestellt, wenn sich der HTML Client gegenüber der zenon Runtime authentifizieren kann. Dies kann über Benutzernamen und Passwort auf zwei Arten erfolgen:

- Automatischer Login durch Konfiguration eines Benutzers im Zuge des Deployments.
  Mehr Details hierzu, siehe Kapitel Deployment der Web Engine (auf Seite 26).
- Manueller Login durch den zenon Web Client in einem Bild vom Typ *Login*.
  Mehr Details hierzu, siehe Kapitel Bild vom Typ *Login* anlegen.

Hinweis: Die Authentifizierung kann durch Übertragung der Login-Daten (Benutzername und Passwort) für einen zenon Benutzer oder einen Active Directory Benutzer erfolgen. Die Validierung der Benutzerdaten erfolgt dabei durch die zenon Runtime.

10 Deployment der Web Engine

Das **zenon Web Engine Deployment Tool** bietet wichtige Operationen zur Verwaltung der Web Engine als Web Applikation in den Internet Information Services (IIS) an.

Um das **Deployment Tool** zu starten:

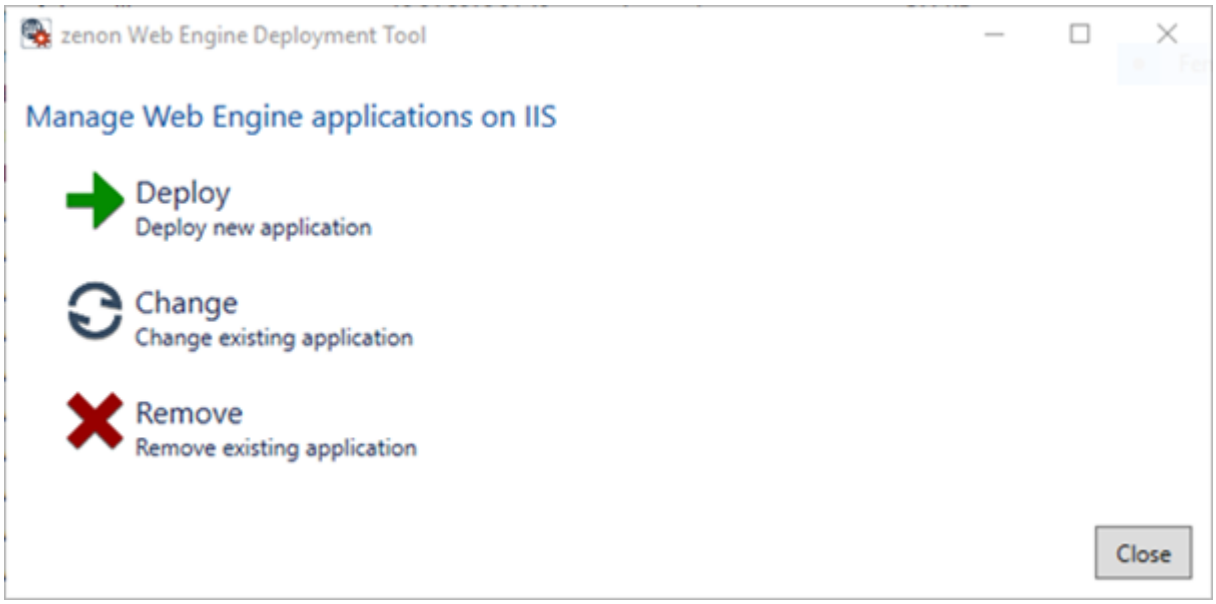
1. Öffnen Sie das zenon **Startup Tool**.
2. Klicken Sie auf den Button **Tools**.
3. Wählen Sie unter **Available applications** das *Web Engine Deployment Tool* aus.
4. Klicken Sie auf **Start**.

   Das Tool wird gestartet

> **Info**
>
> **Hinweis:** Das **Deployment Tool** wird mit dem zenon Web Server automatisch installiert. Für die Verwendung dieses Tools sind Administratorrechte erforderlich.
>
> Das **Deployment Tool** steht nur in englischer Sprache zur Verfügung.

10.1 zenon Web Engine Deployment Tool

Mit dem **zenon Web Engine Deployment Tool** werden HTML Web Engine Instanzen am IIS verwaltet. Sie können neue Instanzen erstellen und bestehende ändern oder entfernen.

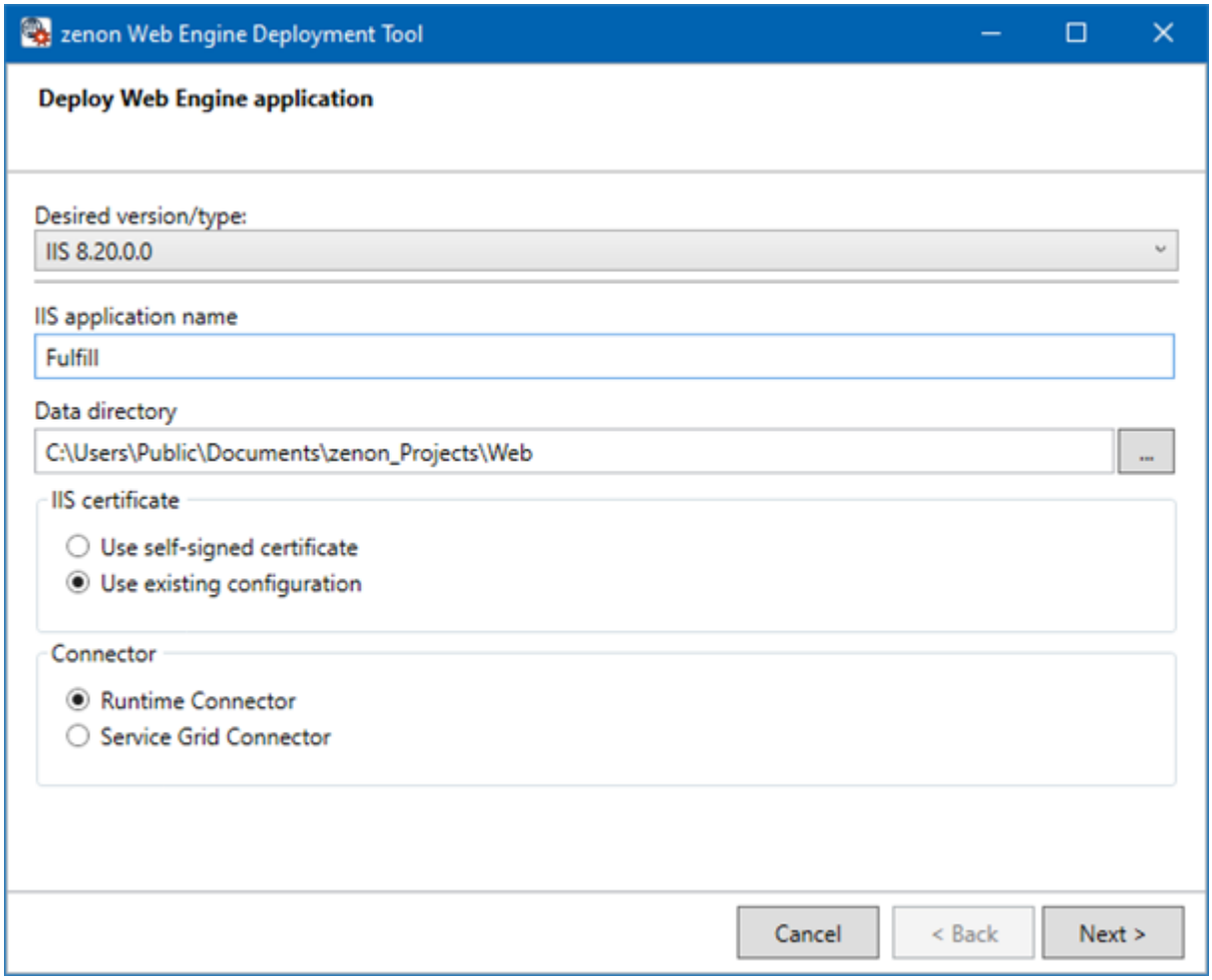


Option	Beschreibung
Deploy	Stellt eine neue Instanz der **HTML Web Engine** am IIS bereit. In der folgenden Registerkarten werden die nötigen Optionen konfiguriert.
Change	Aktualisiert bestehende Web Engine Applications. In der folgenden Registerkarte können für die **HTML Web Engine** geändert werden: ▸ Version. ▸ Konfiguration. ▸ Auswahl des Connectors und dessen Konfiguration.

Option	Beschreibung
	▸ Pfad des Datenordners. Data Directory, in dem die exportierten Projektdaten (***.webx**) liegen. ▸ Sicherheitseinstellungen: Benutzername, Passwort oder Umstellung für automatische Anmeldung. **Hinweis:** Nicht verfügbar, wenn auf dem IIS keine gültige Web Engine Applikation verfügbar ist.
Remove	Entfernt eine laufende Applikation der **HTML Web Engine**. In der folgenden Registerkarte wird die zu entfernende Version ausgewählt. Soll die Web Engine durch eine aktuellere Version ersetzt werden, verwenden Sie die Option **Change**. **Hinweis:** Nicht verfügbar, wenn auf dem IIS keine gültige Web Engine Applikation verfügbar ist.

10.1.1 Allgemeine Einstellungen für die Web Engine

In diesem Dialog konfigurieren Sie allgemeine Einstellungen für den Betrieb der HTML Web Engine.



Welche Optionen verfügbar sind, hängt davon ab, welche Option im Startdialog gewählt wurde:

- **Deploy**: Neukonfiguration einer **HTML Web Engine**.
- **Change**: Änderung einer bestehenden **HTML Web Engine**.
- **Remove**: Entfernen einer bestehenden **HTML Web Engine**.

Option	Beschreibung
Web Engine application to change	Auswahl der Instanz, die geändert werden soll. **Hinweis:** Nur verfügbar bei Auswahl **Change** im Startdialog.
Web Engine application to change	Auswahl der Instanz, die gelöscht werden soll. **Hinweis:** Nur verfügbar bei Auswahl **Remove** im Startdialog.

Option	Beschreibung
Desired version/type	Auswahl der Version aus Dropdownliste. **Hinweis:** Nur verfügbar für **Deploy** und **Change**.
IIS application name	Eingabe des gewünschten Namens für die Web Engine Applikation Dieser Name wird ein Teil der URL sein, unter der die HTML5 Webseite später erreicht wird. Beispiel**: https://Serveradresse/Applikationsname** **Hinweis:** Nur verfügbar für **Deploy** und **Remove** (nur Anzeige).
Data directory	Ordner, aus dem die Web Engine die exportierten Projektdaten auslesen soll. **Hinweis:** In diesem Ordner muss die vom **Web Engine Compiler** generierte webx-Datei zur Verfügung stehen. Beim ersten Zugriff der HTML Web Engine wird die erste webx-Datei in die Dateiliste geladen. Wird die von der Web Engine geladene webx-Datei geändert oder gelöscht, startet die Web Engine automatisch neu und lädt wiederum die erste im Ordner befindliche webx-Datei. **Default-Ordner:** *C:\Users\Public\Documents\zenon_Projects\Web* **Hinweis:** Nur verfügbar für **Deploy** und **Change**.

IIS CERTIFICATE

Die Kommunikation der zenon HTML Web Engine erfolgt immer über eine gesicherte (HTTPS, Port 443) Verbindung.

Definieren Sie in diesem Bereich, ob durch das **Deployment Tool** ein selbst-signiertes Zertifikat für die Kommunikation verwendet wird. Ist noch kein selbst-signiertes Zertifikat vorhanden, wird dies durch das **Deployment Tool** erstellt. Dieses Zertifikat wird dem IIS zugewiesen.

Hinweis: Ein Sicherheits-Zertifikat ist zur Verschlüsselung der Kommunikation zwischen dem zenon Web Server und zenon Web Client zwingend erforderlich.

Option	Beschreibung
Use self-signed certificate	Aktivieren Sie diese Option, wenn Sie ein

Option	Beschreibung
	temporäres, selbst-signiertes Zertifikat erstellen lassen möchten. Diese Option wird beim ersten **Deploy** als Default angeboten. **Hinweis:** Falls noch kein Zertifikat am IIS vorhanden ist, ist diese Option zwingend.
Use existing certificate	Wählen Sie diese Option wenn bereits eine gültige Konfiguration vorliegt. Ist bereits eine gültige Konfiguration vorhanden, wird diese Option als Default angeboten. **Hinweis:** Es wird empfohlen, ein offizielles Zertifikat einer Zertifizierungsstelle zu verwenden. **Mögliche Zertifizierungsstelle:** **https://www.digicert.com/ssl-certificate-installation-microsoft-iis-8.htm**

CONNECTOR

Auswahl des Connectors (auf Seite 22), über den die Verbindung hergestellt wird.

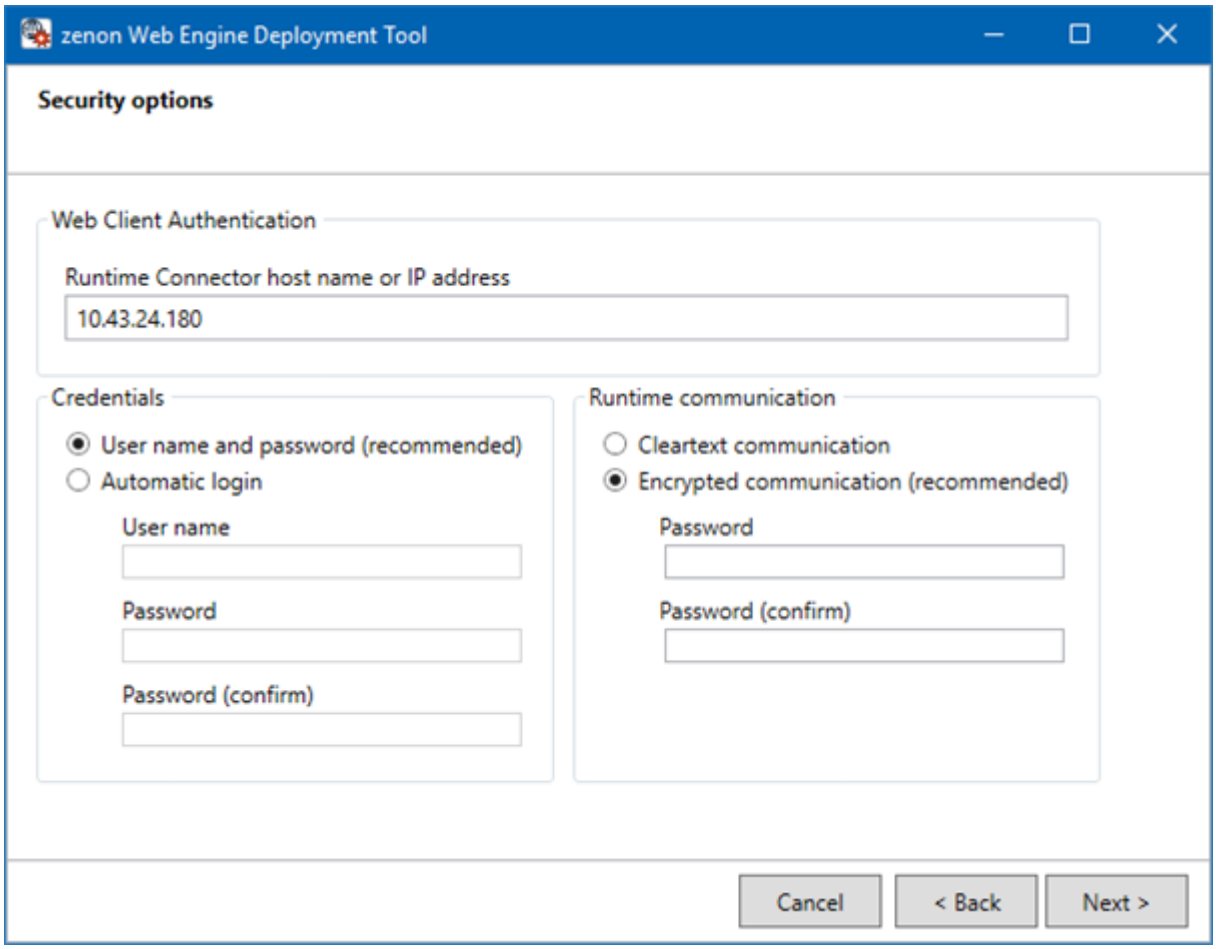
Option	Beschreibung
Runtime Connector	Die Verbindung wird über den Runtime Connector hergestellt.
Service Grid Connector	Die Verbindung wird über den Service Grid Connector hergestellt.

NAVIGATION

Cancel	Verwirft Änderungen und schließt den Dialog.
Back	Schaltet im Ablauf des Tools eine Registerkarte zurück.
Next	Schaltet im Ablauf des Tools eine Registerkarte weiter.

10.1.2 Einstellungen für den Runtime Connector

In diesem Dialog konfigurieren Sie die Einstellungen für den Betrieb der HTML Web Engine über den Runtime Connector.
Nach der Konfiguration erfolgt in einer eigenen Registerkarte die Validierung (auf Seite 37) der Einstellungen.



WEB CLIENT AUTHENTICATION

Einstellungen für die Verbindung.

Option	Beschreibung
Application version	zenon Version der Instanz (nur Anzeige). **Hinweis:** Nur verfügbar für **Remove**.
Runtime Connector host name or IP address	Geben Sie den Host-Namen oder die IP-Adresse des Rechners ein, auf dem die zenon Runtime und der SCADA Runtime Connector installiert sind.

Option	Beschreibung
	Hinweis: Nur verfügbar für **Deploy** und **Change**.

CREDENTIALS

Definieren Sie in diesem Bereich, wie die Authentifizierung des zenon Web Clients erfolgen soll.

Option	Beschreibung
User name and password (recommended)	Die Authentifizierung des zenon Web Clients erfolgt durch manuelle Eingabe von Benutzername und Passwort. **Hinweis:** Die Eingabe von Benutzername und Passwort muss in einem zenon Bild vom Typ **Login** erfolgen.
Automatic login	Aktivieren Sie diese Option, wenn Sie möchten, dass die Web Engine automatisch eine Verbindung zur zenon Runtime aufbaut. Zur Authentifizierung werden die angegebenen Benutzerdaten verwendet. **Achtung:** Bei Verwendung dieser Option erhält jeder zenon Web Client eine Verbindung zur zenon Runtime.
User name	Eingabefeld für zenon Benutzername. Geben Sie hier den gewünschten Benutzernamen ein.
Password	Eingabefeld für Passwort des zenon Benutzers. Geben Sie hier das Benutzer-Passwort ein. **Hinweis:** Nicht verfügbar, wenn für **Web Client Authentication** *User name and password* gewählt wurde.
Password (confirm)	Geben Sie hier das Benutzer-Passwort erneut ein. **Hinweis:** Nicht verfügbar, wenn für **Web Client Authentication** *User name and password* gewählt wurde.

RUNTIME COMMUNICATION

Konfiguration der Kommunikation zur Web Engine.

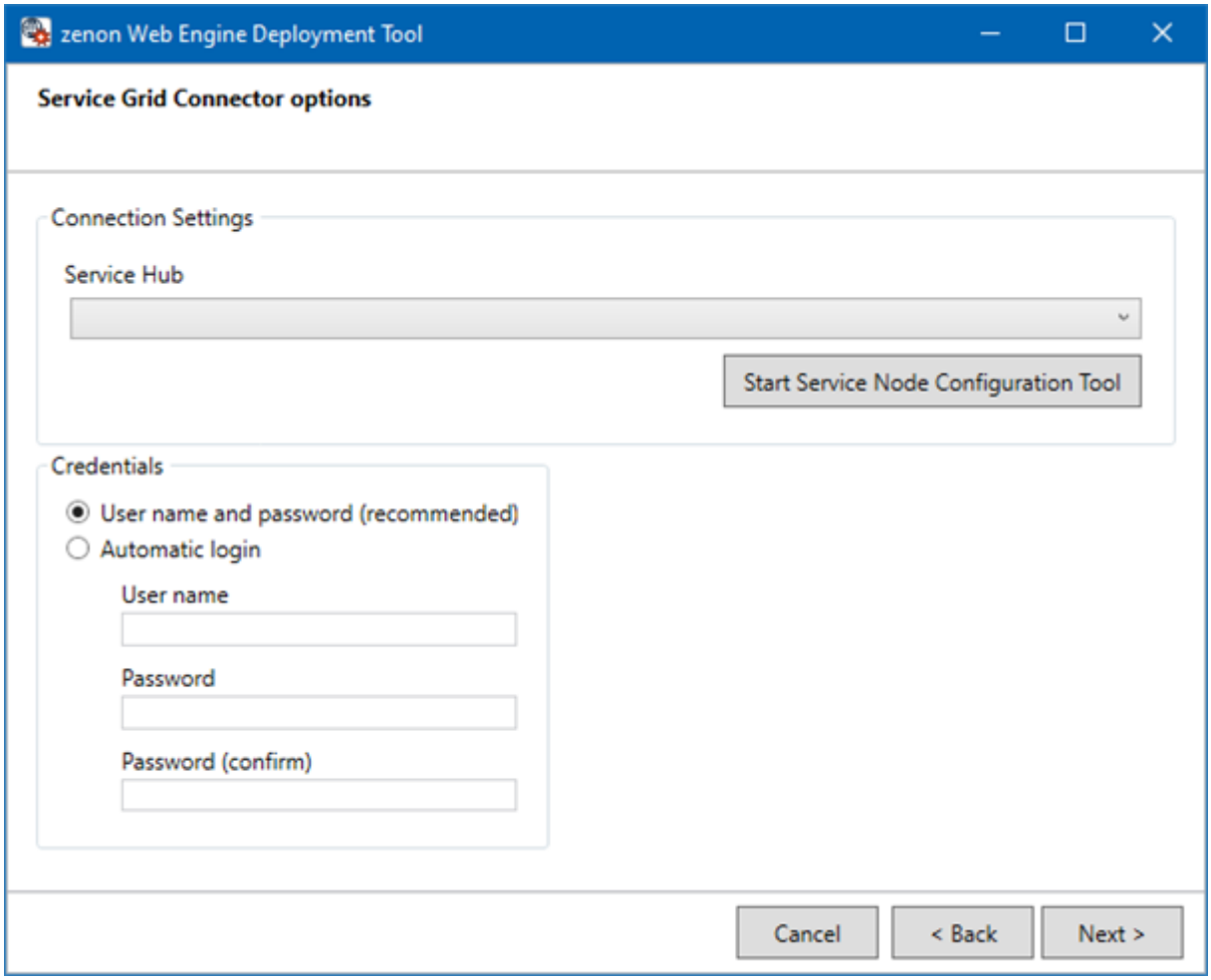
Option	Beschreibung
Cleartext communication	Kommunikation zur Web Engine erfolgt via Klartext ohne Verschlüsselung.
Encrypted communication (recommended)	Die Kommunikation zur Web Engine erfolgt verschlüsselt. **Hinweis:** Die Einstellungen für die Runtime werden im **Startup Tool** in der Registerkarte **Network configuration** mit der Eigenschaft **Encrypt Runtime Connector communication** konfiguriert. Die Passwörter für Runtime und HTML Web Engine müssen übereinstimmen.
Password	Eingabefeld für Passwort für gesicherte Kommunikation. Geben Sie hier das Benutzer-Passwort ein.
Password (confirm)	Geben Sie hier das Benutzer-Passwort erneut ein.

NAVIGATION

Cancel	Verwirft Änderungen und schließt den Dialog.
Back	Schaltet im Ablauf des Tools eine Registerkarte zurück.
Next	Schaltet im Ablauf des Tools eine Registerkarte weiter.

10.1.3 Einstellungen für den Service Grid Connector

In diesem Dialog konfigurieren Sie die Einstellungen für den Betrieb der HTML Web Engine über den **Service Grid Connector** (auf Seite 24).
Nach der Konfiguration wird über die Schaltfläche **Next** das Deployment gestartet (auf Seite 39).



CONNECTION SETTINGS

Einstellungen für die Verbindung zum Service Grid.

Option	Beschreibung
Service Hub	Auswahl des **Service Hub** aus der Dropdownliste. Es werden alle konfigurierten Service Hubs angeboten.
Start Service Node Configuration Tool	Klick auf die Schaltfläche öffnet das **Service Node Configuration Tool**. Damit kann die Verbindung zu

Option	Beschreibung
	einem zusätzlichen **Service Hub** konfiguriert werden, falls der gewünschte **Service Hub** nicht in der Dropdownliste aufscheint

CREDENTIALS

Definieren Sie in diesem Bereich, wie die Authentifizierung des zenon Web Client erfolgen soll.

Option	Beschreibung
User name and password (recommended)	Die Authentifizierung des zenon Web Client erfolgt durch manuelle Eingabe von Benutzername und Passwort. **Hinweis:** Die Eingabe von Benutzername und Passwort muss in einem zenon Bild vom Typ **Login** erfolgen.
Automatic login	Aktivieren Sie diese Option, wenn Sie möchten, dass die Web Engine automatisch eine Verbindung zur zenon Runtime aufbaut. Zur Authentifizierung werden die angegebenen Benutzerdaten verwendet. **Achtung:** Bei Verwendung dieser Option erhält jeder zenon Web Client eine Verbindung zur zenon Runtime.
User name	Eingabefeld für zenon Benutzername. Geben Sie hier den gewünschten Benutzernamen ein.
Password	Eingabefeld für Passwort des zenon Benutzers. Geben Sie hier das Benutzer-Passwort ein. **Hinweis:** Nicht verfügbar, wenn für **Web Client Authentication** *User name and password* gewählt wurde.

Option	Beschreibung
Password (confirm)	Geben Sie hier das Benutzer-Passwort erneut ein. **Hinweis:** Nicht verfügbar, wenn für **Web Client Authentication** *User name and password* gewählt wurde.

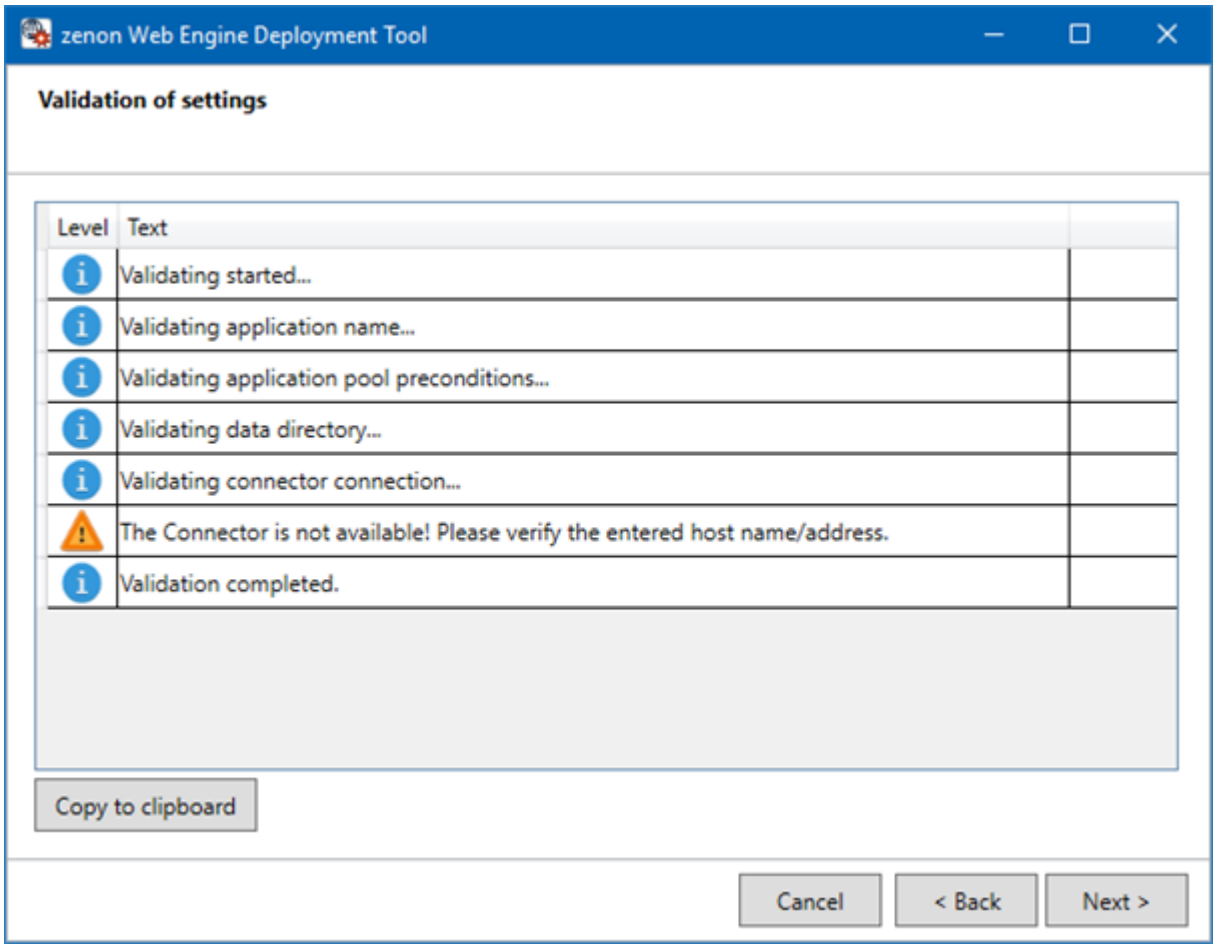
NAVIGATION

Cancel	Verwirft Änderungen und schließt den Dialog.
Back	Schaltet im Ablauf des Tools eine Registerkarte zurück.
Next	Schaltet im Ablauf des Tools eine Registerkarte weiter.

10.1.4 Validierung der Einstellungen

In diesem Dialog werden die Einstellungen validiert. Während der Validierung wird der Fortschritt mit einem grünen Balken visualisiert.

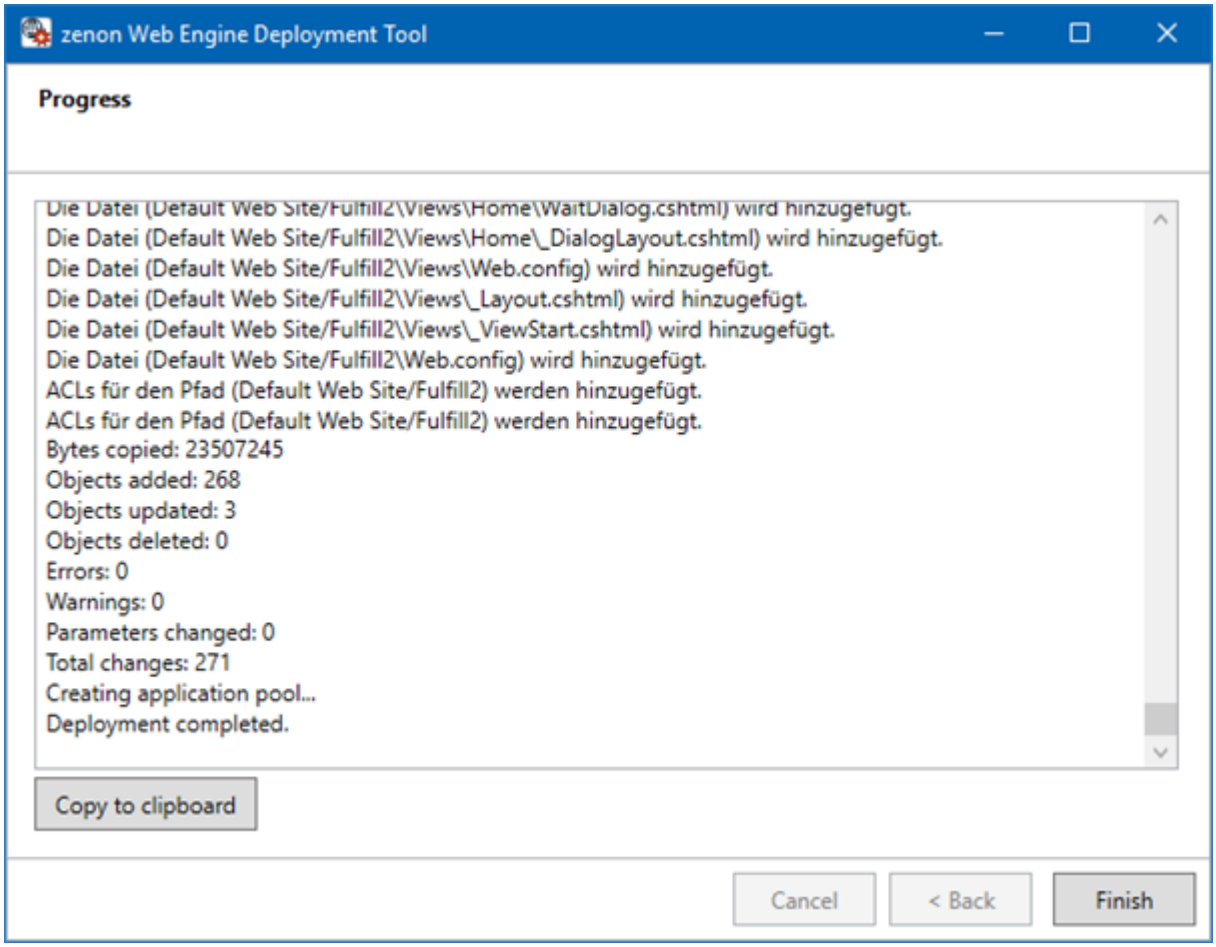
Das Ergebnis der Validierung wird in einer Liste angezeigt.



Mit Klick auf die Schaltfläche **Copy to clipboard** kopieren Sie die Ausgabe in die Zwischenablage.

10.1.5 Fortschritt und Ergebnis

In diesem Dialog sehen Sie den Fortschritt des **Web Engine Deployments** am IIS.



Mit Klick auf die Schaltfläche **Copy to clipboard** kopieren Sie die Ausgabe in die Zwischenablage.

Klicken Sie nach erfolgreichem Abschluss des Vorgangs auf die Schaltfläche **Finish**, um das Deployment Tool zu schließen.

11 Projektieren im Editor

Die notwendigen Daten für die Visualisierung durch die Web Engine werden aus einem zenon Standard Projekt erstellt. Dafür muss der zenon Editor geöffnet sein.

Hinweis: Beachten Sie bei der Projektierung die unterstützten Eigenschaften, Bildelemente und Funktionen der HTML Web Engine.

PROJEKT ANLEGEN

Für die HTML Web Engine Visualisierung mit dem zenon Web Server, legen Sie ein zenon Standard Projekt an. Bestimmte Ressourcen aus einem Globalprojekt können Sie ebenfalls dafür verwenden.

> **Achtung**
>
> Für die HTML Web Engine darf nur der Bildtyp *Standard* als Startbild verwendet werden. Spezielle Bildtypen eignen sich nicht.

11.1 Bild vom Typ HTML anlegen, anpassen und aufrufen

Legen Sie ein Bild vom Typ *HTML* an und erstellen Sie eine Bildumschaltfunktion.

BILD VOM TYP HTML ANLEGEN

Projektierung

Zum Anlegen eines Bildes stehen zwei Vorgehensweisen zur Verfügung:

- die Verwendung des Bilderstellungsdialogs
- die Erstellung eines Bildes über die Eigenschaften

Schritte zum Anlegen des Bildes über die Eigenschaften, wenn der Bilderstellungsdialog in der Menüleiste unter **Extras**, **Einstellungen** und **Assistenten verwenden** deaktiviert wurde:

1. Erstellen Sie ein neues Bild.

   Wählen Sie dazu in der Symbolleiste oder im Kontextmenü des Knotens **Bilder** den Befehl **Bild neu** aus.

2. Ändern Sie die Eigenschaften des Bildes:

   a) Benennen Sie das Bild in der Eigenschaft **Name**.

   b) Wählen Sie in der Eigenschaft **Bildtyp** *HTML* .

   c) Wählen Sie in der Eigenschaft **Schablone** die gewünschte Schablone.

3. Projektieren Sie die Inhalte des Bildes:

   a) Wählen Sie in der Menüleiste den Menüpunkt **Elemente (Bildtyp).**

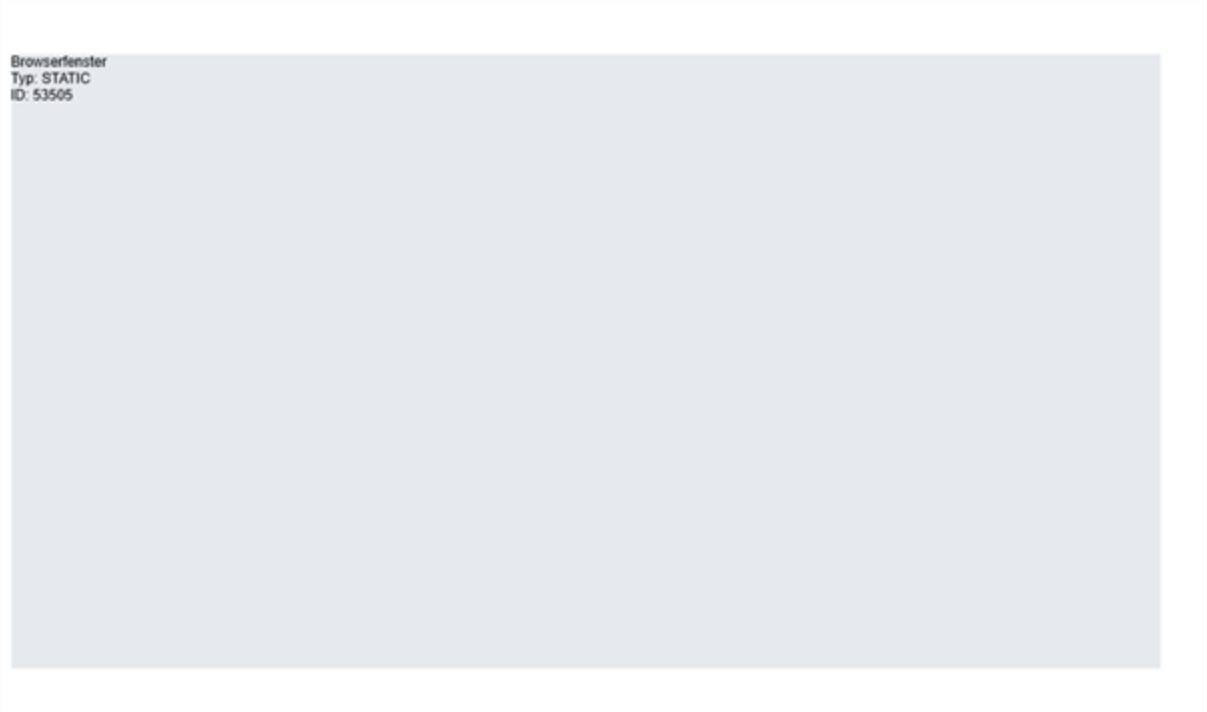
   b) Wählen Sie in der Dropdownliste *Vorlage einfügen*.
   Der Dialog zur Auswahl vordefinierter Layouts wird geöffnet. Damit werden bestimmte Kontrollelemente an vordefinierten Positionen in das Bild eingefügt.

   c) Entfernen Sie nicht benötigte Elemente aus dem Bild.

d) Wählen Sie nach Bedarf zusätzliche Elemente in der Dropdownliste **Elemente** aus. Platzieren Sie diese an der gewünschten Position im Bild.

4. Erstellen Sie eine Bildumschaltfunktion.

BILD VOM TYP HTML



Kontrollelement	Beschreibung
Vorlage einfügen	Öffnet den Dialog zur Auswahl einer Vorlage für den Bildtyp. Vorlagen werden mit zenon mitgeliefert und können auch individuell selbst erstellt werden. Vorlagen fügen vordefinierte Kontrollelemente an vorgegebenen Positionen im Bild ein. Nicht benötigte Elemente können nach dem Anlegen auch einzeln entfernt werden. Zusätzliche Elemente werden aus der Dropdownliste ausgewählt und im zenon Bild positioniert. Elemente können im Bild verschoben und individuell angeordnet werden.
Browser	Kontrollelemente für den Browser.
Browserfenster	Anzeige des Browsers.
Adressfeld	Feld zur Eingabe der Adresse (URL).
Startseite	Aufruf der Startseite.

Kontrollelement	Beschreibung
Vorwärts	Vorwärts blättern.
Zurück	Rückwärts blättern.
Aktualisieren	Anzeige aktualisieren.
Stopp	Navigation beenden.
Suche	Kontrollelemente für die Suche. Beim Klicken auf einen Link im Feld **Suche** wird die entsprechende Seite im Browser angezeigt. So kann im Feld **Suche** z. B. eine Navigationsleiste oder das Ergebnis einer Suchmaschinenabfrage angezeigt werden, ohne dass dessen Inhalt geändert wird, wenn ein Link angeklickt wird.
Suchfenster	Anzeige der Suche.
Suchfeld	Suchen nach Adressen oder Datei.
Startseite	Im Suchbereich zur Startseite zurück.
Vorwärts	Im Suchbereich vorwärts blättern.
Zurück	Im Suchbereich rückwärts blättern.
Aktualisieren	Anzeige im Suchbereich aktualisieren.
Stopp	Suchen abbrechen.
Filter	Filterbox aufschalten.

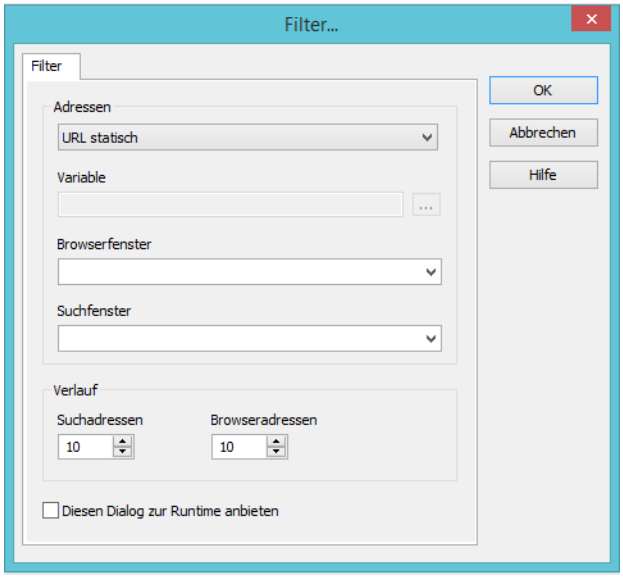
BILD ANPASSEN

Um das Bild an die Nutzung mit der HTML Web Engine anzupassen:

1. Aktivieren Sie für das Bild die Eigenschaft **Im Web verfügbar** unter **Allgemein**.
2. Beachten Sie die **Optionen für das Öffnen externer Webseiten** (auf Seite 43).

BILDUMSCHALTUNG ANLEGEN

Legen Sie eine Bildumschaltfunktion an, um das Bild in der Runtime aufrufen zu können.



Die HTML Web Engine unterstützt folgende Optionen:

Option	Beschreibung
Adressen	▸ *URL statisch* Die URL wird statisch festgelegt. **Hinweis:** Diese Einstellung ist zwingend.
Browserfenster	Geben Sie hier die vollständige URL der externen Webseite ein, inklusive **https://** **Beispiel:** **https://Serveradresse**

11.1.1 Optionen für das Öffnen externer Webseiten

Das Öffnen externer Webseiten wird von der HTML Web Engine entweder über ein eingebettetes Browser-Fenster oder als neues Browser-Fenster unterstützt.

EINGEBETTETES BROWSER-FENSTER:_

Um in der Runtime ein eingebettetes Browser-Fenster von der URL zu sehen, die Sie in der Bildumschaltfunktion eingegeben haben:

- Wählen Sie im Menü **Elemente [Bildtypname]** die Aktion **Browser -> Browser Fenster**.

- Zeichnen Sie im Bild den Rahmen für das Browser-Fenster. Innerhalb dieses Rahmens erfolgt die Darstellung der angegebenen Webseite (URL).

Hinweis: Das Öffnen der eingebetteten Darstellung (**iFrame**) funktioniert nur, wenn die X-Frame-Optionen auf dem Remote Server entsprechend konfiguriert sind. Zusätzlich muss durch die HTTPS-Verbindung zwischen HTML Web Engine und HTML Web Client die Adresse der eingebetteten Webseite ebenfalls für HTTPS zur Verfügung stehen.

NEUES BROWSER-FENSTER:

Um eine externe Webseite in einem neuen Browser-Fenster zu öffnen:

- Löschen Sie das Browser-Fenster Control aus der eingefügten Vorlage im Bild.

  Die externe Webseite wird in einem neuen Browser-Fenster geöffnet, wenn sich kein Browser-Fenster Kontrollelement im Bild befindet.

Hinweis: Das Aufschalten mehrerer URLs wird per Skript unterstützt. Einige Browser verhindern jedoch durch Pop-Up Blocker das Aufschalten von Tabs. In solchen Fällen erscheint ein Dialog mit den URLs, die nicht aufgeschaltet werden konnten. Diese URLs können manuell geöffnet werden.

11.2 Beispiel: Einfaches Startbild

Überprüfen Sie an einem einfachen Beispiel die Funktionsfähigkeit der HTML Web Engine.

Dazu:

1. Erstellen Sie ein Bild vom Typ *Standard*.
2. Aktivieren Sie für das Bild die Eigenschaft **Im Web verfügbar** unter **Allgemein**.
3. Geben Sie dieses Bild in den Projekteigenschaften in der Gruppe **Grafische Ausprägung** als **Startbild** an.

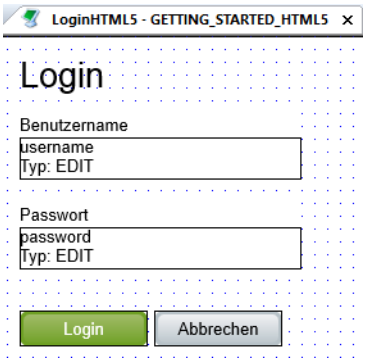
   Hinweis: Mit einem Autostart-Skript können Sie eine beliebige Startseite für das Web definieren.
4. Fügen Sie einfache Elemente in das Bild ein, wie zum Beispiel **Rechteck**, **Kreis** oder **Statischen Text**.
5. Stellen Sie sicher, dass das aktuelle Projekt als Startprojekt gesetzt ist.
6. Exportieren Sie das Projekt für die HTML5 Visualisierung.

ERWEITERUNG: AUTHENTIFIZIERUNG MIT LOGIN

Um Daten mit der zenon Runtime austauschen zu können, muss der zenon Web Client gegenüber der Runtime als Benutzer authentifiziert werden. Mehr Details hierzu lesen Sie im Kapitel: **Client Authentifizierung für eine Verbindung der Runtime** (auf Seite 26).
So erweitern Sie Projekt für eine manuelle Authentifizierung:

1. Fügen Sie ein Dynamisches Element zur Anzeige eines Variablenwerts, zum Beispiel **Dynamischer Text** oder **Zahlenwert** ein.
2. Weisen Sie diesem Element eine Variable aus dem Projekt zu.

   Hinweis: Variablenwerte können auch für Positions- oder Farbdynamiken eines Elements verwendet werden.
3. Erstellen Sie ein neues Bild vom Typ *Login*, das Sie in der Visualisierung anzeigen. Verwenden Sie für die Anzeige die Funktion **Bildumschaltung** verwenden.
4. Aktivieren Sie am Bild in der Gruppe **Allgemein** die Eigenschaft **Im Web verfügbar**.
5. Fügen Sie in das *Login*-Bild die Elemente **Benutzername**, **Passwort**, **Login** und **Abbrechen** ein.



6. Führen Sie den HTML Export für das aktuelle Projekt durch.

Der zenon Web Client hat nun die Möglichkeit, durch Eingabe der Benutzerdaten die Authentifizierung durchzuführen. Bei erfolgreicher Authentifizierung erfolgt der Aufbau der Verbindung zur zenon Runtime. Dadurch stehen beispielsweise Variablenwerte für die HTML5 Visualisierung zur Verfügung.

11.3 Unterstützte Funktionalitäten für die HTML Visualisierung

Für die HTML-Visualisierung stehen grundlegende Elemente, Eigenschaften und Funktionen zur Verfügung:

ALLGEMEIN

Funktionalität	Unterstützung für
Variablen	▸ Anzeige von statischen und dynamischen Werten. ▸ **Sollwert setzen** direkt in der HTML Web Engine. ▸ Dynamisieren der Elementanzeige durch Variablenwerte. z.B.: **Sichtbarkeit/Blinken**, Rotation, Positionierung, Größenanpassung... ▸ Grenzwerte für die Dynamisierung der Element-Anzeige

Funktionalität	Unterstützung für
	(wie **Grenzwertfarbe**, **Grenzwerttext**) werden grundlegend unterstützt. ▸ Permanentes Lesen (auf Seite 68) von Variablen, die bei Grenzwertverletzungen eine zenon Funktion ausführen. **Achtung:** Grenzwertinformationen von Reaktionsmatrizen werden nicht unterstützt! Archive und Variablen können auch über das **Service Grid** bezogen werden.
Schriftarten und Schriftlisten	▸ Auswahl und Darstellung beliebiger Schriftarten, die am System verfügbar sind. **Hinweis:** Gewählte Schriftarten müssen sowohl am Projektierungsrechner als auch auf den zenon Web Clients zur Verfügung stehen. Die Schriften müssen im lokalen Projekt definiert sein. ▸ Darstellung im Schriftstil *Normal*, *Kursiv* und *Fett*. ▸ Auswahl und Darstellung der Schriftart gemäß Schriftliste. ▸ Online Umschaltung der Schriftliste. **Hinweis:** Bei Sitzungsstart wird die erste Schriftliste des zenon Projekts dargestellt

VERKNÜPFUNGEN ERSETZEN

Das Ersetzen von Verknüpfungen erfolgt analog zur Substituierung in der zenon Runtime.

Achtung

Einschränkung bei der Substituierung von Variablen für Symbole:

Wenn Variablennamen sich nur durch Groß-/Kleinschreibung unterscheiden, können bei der Substituierung mehrere Zielvariablen gefunden werden, die dem Kriterium entsprechen. Dies ist dann der Fall, wenn die Option **Groß-/Kleinschreibung beachten** deaktiviert wird.

In Editor und Runtime wird in diesem Fall eine der Variablen nach dem Zufallsprinzip ausgewählt. Diese Auswahl kann in der HTML Web Engine nicht reproduziert werden.

Verhalten beim Kompilieren der Dateien für das Web:

- Werden mehrere mögliche Zielvariablen gefunden, dann schlägt das Kompilieren fehl. Es wird eine Fehlermeldung ausgegeben.
- Wird keine mögliche Zielvariable gefunden, wird eine Warnung ausgegeben.

Tipp: Sorgen Sie in der Projektierung immer für eindeutige Variablennamen, die nicht von der Unterscheidung zwischen Großschreibung und Kleinschreibung abhängen.

GLOBALPROJEKT

Der HTML-Export berücksichtigt die Verwendung folgender Ressourcen aus einem Globalprojekt:

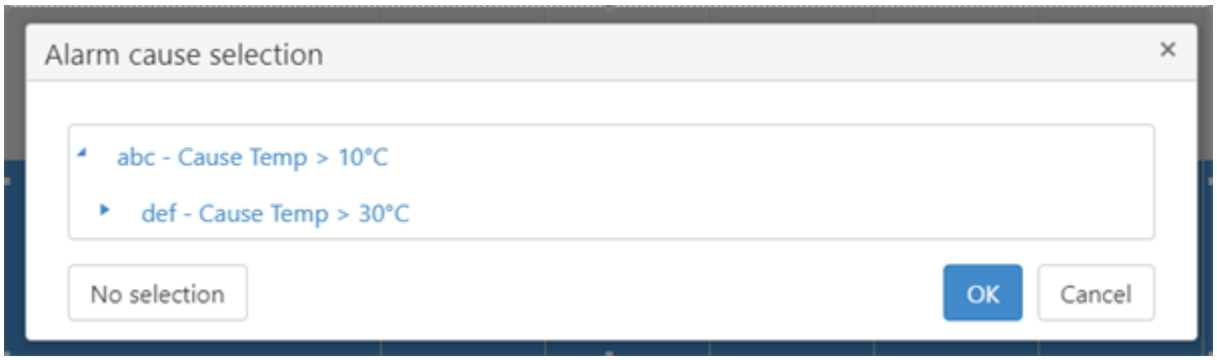
- **Schablonen**
- **Farbpaletten**
- **Sprachdateien**

11.3.1 AML und CEL

Die Bildtypen *AML* und *CEL* unterstützen für die HTML Web Engine folgende Funktionalitäten:

ALARME: ALARMURSACHE ERFORDERLICH

Ist bei Grenzwerten die Eigenschaft **Alarmursache erforderlich** aktiviert, kann in der Web Engine die vorkonfigurierte Alarmursache ausgewählt werden.
Bei erforderlicher Alarmursache wird der Dialog zur Auswahl aufgeschaltet:



Option	Beschreibung
Liste Alarmursachen	Vorkonfigurierte Kontextliste mitz möglichen Alarmursachen. Auswahl durch Klick auf einen Eintrag.
No selection	Hebt Auswahl auf.
OK	Übernimmt Auswahl und schließt den Dialog.
Cancel	Verwirft Auswahl und bricht Dialog ab.

LISTENDARSTELLUNGEN

Listen werden in Seiten unterteilt. Unter der Liste wird eine Fußzeile zur Navigation eingeblendet. Per Klick auf die Symbole können die Einträge der Liste aufgerufen werden. Durch Klick auf die Dropdownliste kann die Anzahl der dargestellten Listeneinträge definiert werden.

SORTIEREN IN DER RUNTIME

Listen in AML und CEL können sortiert angezeigt werden.
Um die Sortierung einer Liste zu ändern, klicken Sie auf den Spaltentitel.
Klick schaltet um zwischen:

- Aufsteigend

- Absteigend
- Standard (wie von der zenon Runtime geliefert)

Das Verhalten der Kopfzeile, der Sortierung und die Spaltenbreiten werden im zenon Editor in den Projekteigenschaften konfiguriert.

SORTIERUNG VON TEXT

Text wird entsprechend der natürlichen Sortierung sortiert:

- alphabetische Reihenfolge
- mehrstellige Zahlen werden als ein Zeichen angesehen

Beispiel:

- Alphabetische Sortierung: 1, 11, 2, 3, 33, 4
- Natürliche Sortierung: 1, 2, 3, 4, 11, 33

SPALTENBESCHRIFTUNG

Die Spalten **Kennung** und **Betriebsmittelkennung** werden mit der Beschriftung **Identification** und **Resource Label** angezeigt. Der in diesen Spalten verwendete Text ist sprachumschaltbar.

SPRACHUMSCHALTUNG IN DER RUNTIME

Für folgende Spalten der AML und CEL steht Sprachumschaltung zur Verfügung:

- **Text**
- **Kennung**
- **Betriebsmittelkennung**
- **Maßeinheit**

Hinweis: Die Sprachumschaltung wird für jede Zelle einzeln durchgeführt. Wird in der Runtime umgeschaltet, kann die Umschaltung auf die neue Sprache einige Sekunden in Anspruch nehmen.

ZEITFORMATE

Datum und Zeit in AML und CEL werden in der lokalisierten Darstellung des jeweiligen Clients angezeigt. Dabei wird die **UTC-DateTime** übertragen und auf Basis der Einstellungen des lokalen Rechners neu formatiert.

Weitere Informationen und Beispiele finden Sie im Handbuch **Runtime**, im Kapitel **Behandlung von Zeitangaben**.

11.3.1.1 Eigenschaften und Optionen

Die HTML Web Engine unterstützt folgende Eigenschaften und Optionen bei Kontrollelementen, Bildumschaltung und Projekteigenschaften:

BILDUMSCHALTFUNKTION AML UND CEL

Registerkarte	Gruppe	Einstellungen und Anmerkungen
Allgemein	**Variablenfilter**	▸ **Variablenname** ▸ **Kennung** **Hinweis:** Groß- und Kleinschreibung wird nicht berücksichtigt.
	Alarmart (nur AML)	Optionen: ▸ **Nur nicht quittierte Alarme** ▸ **Nur gegangene Alarme** ▸ **Nur anstehende Alarme** **Hinweis:** Andere Einstellungen werden ignoriert.
	Datenherkunft	Einstellungen werden ignoriert. Es werden immer **Historische Daten** der zenon Runtime verwendet. Maximum: 65535
	Runtime-Einstellungen	Einstellungen werden ignoriert. Es wird immer die Option **Liste angehalten aufschalten** benutzt. Es werden die Listeneinträge angezeigt, die zum Zeitpunkt Bildaufschaltung in der zenon Runtime vorhanden waren.
Zeit	**Filter**	Optionen: ▸ **Kein Zeitfilter** ▸ **Absoluter Zeitraum** ▸ **Relativer Zeitraum**
	Einstellungen	Nur die Option **Vorgabe**. Alle anderen Einstellungen werden ignoriert.

Registerkarte	Gruppe	Einstellungen und Anmerkungen
Spalteneinstellungen	**Spalten**	▸ **Alarmzustand** (Inklusive Anzeige über Kreissymbol oder Grafikdatei.) ▸ **Zeit kommt** ▸ **Zeit geht** (nur AML) ▸ **Zeit quittiert** (nur AML) ▸ **Text** ▸ **Variablenname** ▸ **Wert** ▸ **Maßeinheit** ▸ **Benutzer vollständiger Name** ▸ **Computername** ▸ **Kommentar** **Hinweis:** Die eingestellte Darstellungsreihenfolge wird berücksichtigt. Die Spaltenbeschreibung ist editierbar und sprachumschaltbar.
	Tabelleneinstellungen	**Immer aktiv:** ▸ **Abwechselnde Hintergrundfarben verwenden** ▸ **Raster anzeigen** ▸ **Absteigend sortieren** **Hinweis:** Farbpaletten-Umschaltung wird unterstützt für: ▸ **Zeilenfarbe 1** ▸ **Zeilenfarbe 2**
Anlagenmodellierung		Aus Lokal- und Globalprojekt.

KONTROLLELEMENTE FÜR BILDTYP ALARMMELDELISTE

Gruppe	Untergruppe	Einstellungen und Anmerkungen
Kopfzeile und Raster	**Kopfzeile**	**Kopfzeile anzeigen**: Einstellung immer *aktiv*. **Anzeigestil**: Einstellung immer *Standard*. **Füllfarbe**: Ergibt Hintergrundfarbe. Wird auch für die Fußzeile im Web verwendet für: ▸ Statische/Fixe Farbe. ▸ Farbpalette und Farbpaletten Umschaltung. **Schrift** wird auch in der Fußzeile im Web angewendet für: ▸ Statische Auswahl. ▸ Schriftlisten Umschaltung. **Textfarbe** wird auch in der Fußzeile im Web angewendet für: ▸ Statische/Fixe Farbe. ▸ Farbpalette und Farbpaletten Umschaltung.
Füllung	**Füllung**	**Textfarbe** wird verwendet für: ▸ Statische/Fixe Farbe ▸ Farbpalette und Farbpaletten Umschaltung
Darstellung	**Darstellung**	Wird verwendet für: ▸ Statische Auswahl ▸ Schriftlisten Umschaltung

PROJEKTEIGENSCHAFTEN AML

Eigenschaftengruppe	Unterstützte Eigenschaften
Alarmmeldeliste	**Kopfzeile AML**

Eigenschaftengruppe	Unterstützte Eigenschaften
Datenspeicherung AML	Nicht verfügbar.
Alarm kommt	Alle.
Alarm geht	Alle.
Alarm quittiert	Alle.
Alarmquittierung bestätigt	Nicht verfügbar. Alarme, deren Quittierung bestätigt wurde, werden in der HTML Web Engine nicht angezeigt.
Alarmstatuszeile	Nicht verfügbar.

11.3.1.2 Alarmquittierung

Alarme können einzeln ausgewählt und quittiert werden. Es werden nur Alarme mit aktivierter Eigenschaft **Quittierpflichtig** unterstützt.

Dabei gilt:

- Die Quittierung des Alarms erfolgt via Button **Quittieren** des Bildtyps *Alarmverwaltung*.
- Ist für einen Grenzwert die Eigenschaft **Quittierpflichtig** nicht aktiviert, wird der Button **Quittieren** im Bildtyp *Alarmverwaltung* deaktiviert.
- Der Button **Alle quittieren** des Bildtyps *Alarmverwaltung* wird nicht unterstützt.
- Das Quittieren eines Alarms setzt für den angemeldeten Benutzer die notwendigen Funktionsberechtigungen voraus.
  Sind diese nicht vorhanden, wird eine Fehlermeldung angezeigt.

  Hinweis: Stellen Sie sicher, dass der Benutzer über die Funktionsberechtigung *Alarm quittieren über Bilder vom Typ Alarmmeldeliste* verfügt.
  Weitere Informationen finden Sie im Handbuch Benutzerverwaltung im Kapitel Funktionsberechtigungen.

11.3.2 Bilder, Schablonen, Elemente und Symbole

Unterstützung für Bilder und Schablonen sowie Elemente und Symbole in Bildern.

BILDER UND SCHABLONEN

Bild/Schablone	Unterstützung für
Schablone	▸ Aufschaltung rechteckiger Schablonen an absoluter Position. **Achtung:** Schablonennamen müssen eindeutig sein. Groß-/Kleinschreibung wird nicht beachtet.
Bilder allgemein	▸ Darstellung von Bildern in der Größe der verknüpften Schablone. ▸ Anzeige von Hintergrundfarbe und Hintergrundbild. ▸ Ausführung einer Bild-Start- und -Ende-Funktion.
Bildtyp Standard	▸ Darstellung von Bildern dieses Typs.
Bildtyp Login	▸ Darstellung von Bildern dieses Typs. ▸ Benutzer Authentifizierung mit den Bildtyp-spezifischen Elementen **Eingabe Benutzername** und **Eingabe Passwort** oder **Login**-Befehl.
Bildtyp AML (auf Seite 47)	▸ Darstellung von Bildern dieses Typs. ▸ Darstellung von dynamischen AML-Listen. Dynamisch bedeutet, dass die dargestellten Daten aktualisiert werden.
Bildtyp CEL (auf Seite 47)	▸ Darstellung von Bildern dieses Typs. ▸ Darstellung von dynamischen CEL-Listen. Dynamisch bedeutet, dass die dargestellten Daten aktualisiert werden.
Bildtyp HTML (auf Seite 40)	▸ Darstellung von Bildern dieses Typs. ▸ Darstellung der Bildtyp-spezifischen Elemente wie Web Browser
Bildtyp ETM (auf Seite 58)	▸ Darstellung von Bildern dieses Typs. ▸ Statische Darstellung von einfachen Liniendiagrammen. Statisch bedeutet, dass die dargestellten Daten nicht aktualisiert werden.

BILDELEMENTE

STATISCHE BILDELEMENTE

Unterstützt werden:

Element	Unterstützung für
Kreis	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Eigenschaft **Farbverlauf invertiert**. Bei Auswahl *Farbverlauf* für die Eigenschaft **Füllmuster** wird der Helligkeitsverlauf des Füllmusters invertiert dargestellt.
Kreisbogen	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken.
Kreissegment	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Eigenschaft **Farbverlauf invertiert**. Bei Auswahl *Farbverlauf* für die Eigenschaft **Füllmuster** wird der Helligkeitsverlauf des Füllmusters invertiert dargestellt.
Linie	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken.
Polygon	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Eigenschaft **Farbverlauf invertiert**. Bei Auswahl *Farbverlauf* für die Eigenschaft **Füllmuster** wird der Helligkeitsverlauf des Füllmusters invertiert dargestellt. ▸ **Hinweis:** Die Eigenschaft **Rundung** wird nicht unterstützt.
Polylinie	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und

Element	Unterstützung für
	Positionsdynamiken. ▸ **Hinweis:** Die Eigenschaft **Rundung** wird nicht unterstützt.
Rechteck	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Eigenschaft **Farbverlauf invertiert**. Bei Auswahl *Farbverlauf* für die Eigenschaft **Füllmuster** wird der Helligkeitsverlauf des Füllmusters invertiert dargestellt.
Statischer Text	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken.

DYNAMISCHE BILDELEMENTE ALLGEMEIN

Unterstützt werden:

Element	Unterstützung für
Button	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Ausführung von Funktionen des lokalen Projekts. **Hinweis:** Für die Ausführung der Aktion wird die entsprechende Benutzerebene geprüft. ▸ Eigenschaften **Füllmuster** und **Farbverlauf**: **Farbverlauf** wirkt sich nur dann aus, wenn für **Füllmuster** *Farbverlauf* gewählt wird.
Combi-Element	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Setzen von numerischen Grenzwerten über den Standardialog. ▸ Schalter und Taster.

Element	Unterstützung für
	Details siehe Abschnitt **Combi-Element** (auf Seite 61).
Dynamischer Text	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Anzeige der Variableninformationen *Variablenwert*, **Name**, **Kennung**, **Betriebsmittelkennung**, **Maßeinheit** und **Grenzwerttext** möglich. ▸ **Sollwert setzen** mit Dialog. ▸ **Sollwert setzen** über Element. Details siehe Abschnitt **Sollwert setzen** (auf Seite 63). ▸ **Hinweis:** Für die Ausführung von **Sollwert setzen** wird die entsprechende Benutzerebene geprüft.
Zahlenwert	▸ Elementspezifischen Anzeigeoptionen für Anzeige, Farben und Fülloptionen. Nicht unterstützt werden Effekte. ▸ Darstellungsdynamiken über Variablen für Farbgebung und Positionsdynamiken. ▸ Sollwert setzen mit Dialog unter Berücksichtigung statischer Sollwertgrenzen. **Hinweis:** Für die Ausführung der Aktion wird die entsprechende Benutzerebene geprüft.

ELEMENTGRUPPEN UND SYMBOLE

Unterstützt werden:

Funktionalität	Unterstützung für
Elementgruppe	▸ Darstellung von Elementgruppen. ▸ Die Darstellung der enthaltenen Elemente erfolgt gemäß deren Projektierung und unterstützten Eigenschaften.
Verknüpftes Symbol	▸ Darstellung von verknüpften Symbolen. ▸ Unterstützung von **Verknüpfungen ersetzen** bei Anwendung im Bild. Dabei müssen daraus resultierende Einträge auf Ressourcen im lokalen Projekt verweisen. ▸ Die Darstellung der enthaltenen Elemente erfolgt gemäß

Funktionalität	Unterstützung für
	deren Projektierung und unterstützten Eigenschaften.
Symboleigenschaften	▸ Freigegebene Symboleigenschaften für alle von der HTML Web Engine unterstützten Eigenschaften. Das beinhaltet alle zenon Eigenschaften von Elementen, die von der HTML Web Engine unterstützt werden. Nicht unterstützte Eigenschaften führen beim Kompilieren (auf Seite 70) zu einer Fehlermeldung.
Gruppierung freigegebener Eigenschaften	▸ Gruppen für freigegebene Symboleigenschaften werden übernommen und die enthaltenen Eigenschaften angewendet.

SYMBOLE GLOBALPROJEKT UND LOKALPROJEKT

Symbole können exportiert und verwendet werden aus:

- Globalprojekt
- Lokalprojekt

Symbole aus dem Globalprojekt erhalten beim Kompilieren für das Web das Präfix **(g)**.

Hinweis:

Befindet sich im Lokalprojekt ein Symbol, das beim Export den gleichen Namen hat wie ein Symbol im Globalprojekt, dann wird das Symbol aus dem Lokalprojekt verwendet. Zum Beispiel:

- Globalprojekt: **rotor**
- Lokalprojekt: **(g)rotor**

Beim Export wird im Globalprojekt aus **rotor** => **(g)rotor**. Dieser Name entspricht dem **(g)rotor** aus dem Lokalprojekt. Es wird nur das Symbol aus dem Lokalprojekt verwendet.

11.3.2.1 Bild vom Typ Erweiterter Trend

Der Bildtyp *Erweiterter Trend* unterstützt für die HTML Web Engine folgende Funktionalitäten:

OPTIONEN DER BILDUMSCHALTFUNKTION

Registerkarte	Gruppe	Einstellungen und Anmerkungen
Daten	**Datenherkunft**	Option: nur **Archivdaten**
	Optionen	Die Einstellungen der **Optionen** werden für

Registerkarte	Gruppe	Einstellungen und Anmerkungen
		die HTML Web Engine nicht unterstützt.
	Kurven	Anzeige von einfachen Kurven ohne weiteren Optionen.
Darstellung	**Diagramm-Name**	Darstellung des Diagramm-Namens, wie projektiert
	Aktualisierung	Optionen der Eigenschaftengruppe **Aktualisierung** werden nicht unterstützt.
	Design	Optionen der Eigenschaftengruppe **Design** werden nicht unterstützt. Es wird eine vordefinierte Darstellung angezeigt.
	Runtime	Die Anzeige dieses Dialogs wird in der HTML Web Engine nicht unterstüzt. Eine Filterung ist nicht möglich.
X-Achse	**X-Achse**	Option: ▸ **nur YT-Anzeige** **Hinweis:** XY-Anzeige wird nicht unterstützt.
Zeit	**Filter**	Unterstützt werden: ▸ **Absoluter Zeitraum** ▸ **Relativer Zeitraum** Alle anderen Zeitfilter werden nicht unterstützt. **Achtung:** Bildumschaltungen mit ungültigen Zeitfilterkonfigurationen stehen in der HTML Web Engine nicht zur Verfügung. Buttons mit entsprechenden Aufrufen werden deaktiviert.
Chargen		Die Einstellungen der Registerkarte **Chargen** werden für die HTML Web Engine nicht unterstützt.
Spalteneinstellungen		Die Einstellungen der Registerkarte **Spalteneinstellungen** werden für die HTML Web Engine nicht unterstützt.

Registerkarte	Gruppe	Einstellungen und Anmerkungen
Druckereigenschaften		Die Einstellungen der Registerkarte **Druckereinstellungen** werden für die HTML Web Engine nicht unterstützt.

KONTROLLELEMENTE DIAGRAMM

Gruppe	Untergruppe	Einstellungen und Anmerkungen
Darstellung		Die Konfiguration der Darstellung wird für die HTML Web Engine nicht unterstützt. Das Verhalten ist fix vorgegeben.
Linien		Die Konfiguration der Linien wird für die HTML Web Engine nicht unterstützt. Das Verhalten ist fix vorgegeben.
Füllung		Die Konfiguration von Füllungen wird für die HTML Web Engine nicht unterstützt. Das Verhalten ist fix vorgegeben.

11.3.2.1.1 Datenverdichtung

Für die Darstellung im Erweiterten Trend werden Daten unter bestimmten Bedingungen verdichtet. Damit wird die Schnelligkeit der Darstellung über die HTML Web Engine beschleunigt, so dass sie jener in der zenon Runtime entspricht.

Ablauf:

- Die maximale Anzahl der darzustellenden Variablen wird durch die Breite der Schablone in Pixel bestimmt, auf der das Bild beruht.
- Werden bei einer Archivabfrage weniger Variablen zurückgegeben als Pixel vorhanden sind, erfolgt keine Verdichtung.
- Übersteigt die Anzahl der Archivvariablen die Schablonenbreite, dann wird eine Mittelwertbildung durchgeführt.
  Dazu wird die Zeitachse in Zeitbereiche eingeteilt.
  Die von der Runtime übergebenen Archivwerte werden in die jeweiligen Zeitbereiche eingeordnet. Über die Archivwerte innerhalb eines Zeitbereichs wird der Mittelwert berechnet. Befindet sich innerhalb eines Zeitbereichs kein Archivwert, wird dieser Zeitbereich gelöscht.
  Hinweis: Die Mittelwertbildung wird nicht nur für numerische Variablen durchgeführt, sondern auch bei Binärvariablen angewendet.

- Nicht notwendige Variablen bleiben unberücksichtigt, da diese für die aktuelle Darstellung nicht benötigt werden. Dies reduziert Hintergrundberechnungen, spart Rechnerleistung und beschleunigt die Aufschaltzeit.
  Das passiert unabhängig von der Anzahl der Archivwerte und der Mittelwertbildung.
  Dazu werden 3 zeitlich aufeinanderfolgende Archivwerte einer Variable analysiert. Haben alle 3 Variablen den gleichen Wert, wird die mittlere Variable entfernt.
  Hat z. B. eine aufgezeichnete Archivvariable über den gesamten Abfragezeitraum den gleichen Wert, werden nur 2 Variablen (Anfang und Ende) dargestellt.

Hinweis: Bei einer Mittelwertbildung wird in der Regel die Trendlinie anders dargestellt als bei Rohwerten.

11.3.2.2 Combi-Element

Das Combi-Element unterstützt in der HTML Web Engine:

Gruppe und/oder Eigenschaft	Anmerkung
Bezugspunkt	
Bitmap Einstellungen	
Blinken	**Nicht unterstützt wird:** ▸ **Blinken aller Variablen berücksichtigen**
Darstellung	**Nicht unterstützt werden:** ▸ **Stilgruppe** ▸ **3D** ▸ **Status der Hauptvariable anzeigen** ▸ **Status anzeigen mit Quadrat** ▸ **Darstellungsart** ▸ **Konfiguration und Test** ▸ **Projekteigenschaften bei "gesperrt" übernehmen**
Füllung	**Nicht unterstützt wird:** ▸ **Füllungsstil**
Position	
Schalter	Das Element schaltet den Wert einer Bit-Variable.
Sichtbarkeit	**Nicht unterstützt wird:**

Gruppe und/oder Eigenschaft	Anmerkung
	▸ **Verriegelung**
Sollwert setzen	Numerische Werte über Sollwert-Setzen Dialog und Funktion: Die verknüpfte Funktion wird so wie in der zenon Runtime erst nach dem Schließen des Dialogs zum Sollwert-Setzen ausgeführt. Das Schließen erfolgt durch erfolgreiches Absetzen mit Klick auf **OK** oder durch Abbrechen.
SOLLWERT SETZEN/BINÄRWERT	Die Eigenschaft **Beim Verlassen zurücksetzen** ist immer aktiv. **Nicht unterstützt werden:** ▸ **Gedrückt Zustand halten** ▸ **Sofortreaktion** ▸ **Haltezeit [ms]**
Sollwert setzen/Numerischer Wert	Für Eigenschaft **Sollwert absetzen über** wird nur die Einstellung *Standard-Dialog* unterstützt. **Nicht unterstützt werden:** ▸ **ohne Dialog** ▸ **Externes Programm** ▸ **Keyboardbild verwenden** ▸ **Keyboardbild**
Sollwert setzen/Sollwert-Grenzen statisch	
Taster und **Taster EIN**	Die Eigenschaft **Beim Verlassen zurücksetzen** ist immer aktiviert und kann auch nicht deaktiviert werden.
Text	**Nicht unterstützt wird:** ▸ **Textstil**
Texte gekettet	
Variable/Funktion	**Nicht unterstützt werden:** ▸ **Parameter für Substitution** ▸ **Von aufrufendem Bild übernehmen**

Gruppe und/oder Eigenschaft	Anmerkung
	▸ **Funktionsausführung beim Drücken**
Statusauswahl per Variablenstatus und Textdynamisierung:	Auswahl von Status aufgrund passenden Variablenstatus, zusätzlich zu Variablenwert: **Darstellung Zustandstext:** ▸ **Textfarbe dynamisch** ▸ Text bei gedrückt ▸ **Transparent** ▸ Platzhalter (Groß-/Kleinschreibung wird beachtet): *%n* (Variablenname) *%l* (Variablenkennung) *%v* (Variablenwert) *%u* (Maßeinheit

FORMELN FÜR ZUSTÄNDE

Das Combi-Element unterstützt die Verwendung von Formeln für Zustände. Unterstützt werden alle Wert- und Statusbits für mehrere Quellvariablen, sowie die in der zenon Runtime verfügbaren Operatoren.

Ausnahme: Das Statusbit *unquittiert* wird nicht unterstützt. Es wird in der HTML Web-Engine immer als *0* behandelt.

11.3.2.3 Sollwert setzen

Für die Elemente **Dynamischer Text** und **Zahlenwert** sind für die Eigenschaft **Sollwert absetzen über** die Werte *Standarddialog* oder *Element* verfügbar.

SOLLWERT SETZEN ÜBER ELEMENT

In der Runtime wird die Eingabefunktion aktiviert, wenn auf das Element geklickt wird. Ist ein Text oder ein Vorgabetext vorhanden, wird dieser vorausgewählt. Auf mobilen Geräten wird nach der Aktivierung das Keyboard angezeigt.
Achtung: Bei iOS muss das Element erneut ausgewählt werden, um das Keyboard anzuzeigen.

UNTERSCHIEDE ZENON RUNTIME UND HTML WEB ENGINE

Die Darstellung unterscheidet sich für zenon Runtime und HTML Web Engine:

Vorgabe im Editor für Sollwert setzen über Element	zenon Runtime	HTML Web Engine
Vorgaben: ▸ **Dynamischer Text** und Variable vom Typ *String* ▸ **aktuellen Wert vorschlagen**: *aktiviert*	Kein Wert wird angezeigt.	Der aktuelle Wert wird angezeigt und ausgewählt.
Vorgaben: ▸ **Dynamischer Text** und Variable vom Typ *String* ▸ **aktuellen Wert vorschlagen**: *deaktiviert* ▸ **Sollwert Text**: Text vorhanden	Kein Wert wird angezeigt.	Der Sollwerttext wird angezeigt und ausgewählt.
Vorgaben: ▸ **Dynamischer Text** und Variable vom Typ *String* ▸ **aktuellen Wert vorschlagen**: *deaktiviert* ▸ **Sollwert/Ändern um**: enthält Wert	Der aktuelle Variablenwert wird angezeigt.	Der eingetragene Sollwert wird angezeigt.
Vorgaben: ▸ **Verdeckte Eingabe**: *aktiviert*	Bei der Eingabe wird der im Editor definierte Buchstabe für maskierte Eingabe in einem Textfeld angezeigt.	Bei Eingabe wird das Standardzeichen für maskierte Eingabe in einem Textfeld angezeigt: Punkt oder Stern.

11.3.3 Funktionen

Funktionen und Skripte.

Funktion	Unterstützung für
Bildumschaltung	▸ Aufschaltung von Bildern des lokalen Projekts. ▸ Unterstützung von *Verknüpfungen ersetzen*, wobei resultierende Einträge auf Ressourcen im lokalen Projekt

Funktion	Unterstützung für
	verweisen müssen.
Schablone schließen	▸ Schließen von Schablonen mit dem angegebenen Schablonennamen.
Sollwert setzen/modifizieren	▸ Direktes Setzen von vordefinierten Variablenwerten.
Sprachumschaltung	▸ Online-Umschaltung von **Sprachdatei** und **Schriftliste**.
Farbpalette umschalten	▸ Online Umschaltung der **Farbpallette** zur grafischen Darstellung.
Skript ausführen	▸ Ausführung von Funktionen des lokalen Projekts. Nicht unterstützte Funktionen werden von der Ausführung ausgenommen.
Logout	▸ Abmelden des aktuellen Benutzers einer zenon Web Client Sitzung und Abbau der Verbindung zur zenon Runtime ▸ Die zenon Web Client Sitzung wird im Offline Modus fortgesetzt
Audiodatei ausführen	▸ Spielt eine Audiodatei (auf Seite 67) ab.
Dauerton Start	▸ Startet einen Dauerton (auf Seite 67).
Dauerton Stopp	▸ Stoppt einen Dauerton (auf Seite 67).

11.3.3.1 Automatischer Skript-Aufruf bei Start einer zenon Web Client Sitzung

Beim Starten einer HTML Web Client-Sitzung kann ein Skript automatisch aufgerufen werden. Damit kann zum Beispiel eine spezielle Startseite für die Webanwendung vorgegeben werden.

Für das Skript ist folgender Skript-Name reserviert: *AUTOSTART_HTML_WEBCLIENT*

Hinweis: Dieses Skript wird beim Sitzungsstart jedes HTML Web Client ausgeführt. Der Name des Skriptes darf nicht verändert werden.

11.3.3.2 Individueller Skript-Aufruf mittels URL-Erweiterung bei Start einer zenon Web Client Sitzung

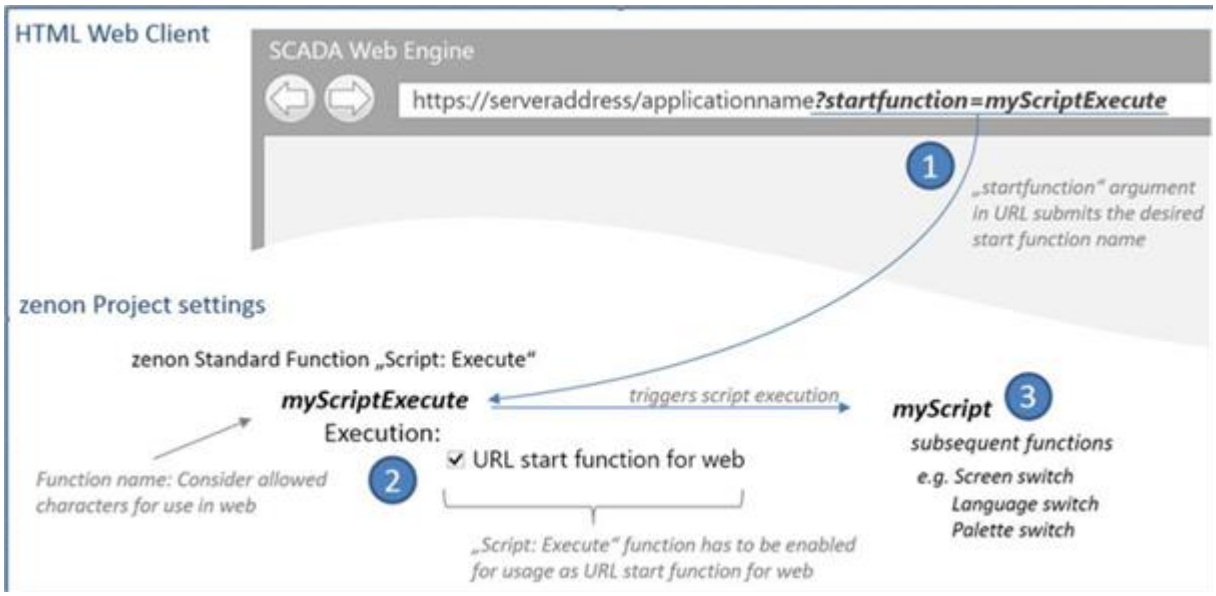
Die HTML Web Engine ermöglicht das individuelle Ausführen von zenon Funktionen im Zuge des Sitzungsstarts für einen HTML Web Client. Die Funktionsausführung erfolgt durch ein Skript, das über

die Funktion **Skript ausführen** gestartet wird. Der Name der gewünschten Startfunktion wird in der URL für den Aufruf der Webseite mit dem Argument **startfunction** übermittelt. Die Funktion zur Ausführung des Skripts muss explizit für den Aufruf als URL-Startfunktion freigegeben werden.

Hinweis: Sämtliche Einstellungen, die für die Nutzung der URL-Startfunktion erforderlich sind, müssen zum Zeitpunkt der Kompilierung für das Web im Projekt vorhanden sein.

Ablauf für die Anwendung der Funktionalität:

1. Der Aufruf der Webseite mit dem zusätzlichen Argument **startfunction** für den Aufruf der gewünschten Funktion **Skript ausführen**.
   Der Aufruf erfolgt mit dem Funktionsnamen.
2. Sofern die Funktion für die Verwendung als URL-Startfunktion freigegeben wurde, erfolgt die Ausführung des zugewiesenen Skripts. Die Ausführung des Skripts erfolgt individuell für die vorliegende HTML Web Client Sitzung. Falls die angesprochene Funktion nicht existiert oder nicht freigegeben wurde, erfolgt der standardmäßige Webseiten-Aufruf. Die Sitzung wird entsprechend dem Aufruf ohne **startfunction**-Argument aufgebaut.



Hinweis: Dieser Screenshot steht nur in englischer Sprache zur Verfügung.

Für die Benennung des Funktionsnamens im Web gelten folgende Einschränkungen:

Erlaubt	Verboten
Alphanumerische Zeichen (0-9, a-z, A-Z)	Umlaute
$ - _ . + ! * ' ( ) ,	ASCII Steuerzeichen (**0x00-0x1F;0x7F**)
	Reservierte Zeichen (**& / : ; ? @**)
	Unklare Zeichen (wie Leerzeichen und **" ' < > # % {**

Erlaubt	Verboten
	} | \ ^ ~ [ ])

11.3.4 Tonsignal oder Dauerton abspielen

In der HTML Web Engine können Tonsignale und Dauertöne abgespielt werden.

- Tonsignal: Eine Audiodatei wird einmal abgespielt.
  Sie wird gestartet mit der Funktion **Audiodatei ausführen**.
- Dauerton: Eine Audiodatei wird permanent abgespielt.
  Sie wird gestartet mit der Funktion **Dauerton Start** und beendet mit der Funktion **Dauerton Stopp**.

> **Achtung**
>
> Audiodateien müssen im Format ***.wav** vorhanden sein.
>
> Diese werden nicht unterstützt für:
>
> - Browser Safari
> - iPad

FUNKTIONEN AUSFÜHREN

Die Funktionen zum Abspielen eines Tonsignals oder zum Starten und Stoppen eines Dauertons können ausgeführt werden über:

- Button, der mit der Funktion verknüpft ist
- Aufruf in einem Skript, das mit einem Button verknüpft ist
- einen Grenzwert, der bei Verletzung die jeweilige Funktion aufruft

REGELN

Beim Abspielen von Tonsignalen und Dauertönen in der HTML Web Engine gilt:

- Es kann immer nur eine Audiodatei gleichzeitig abgespielt werden.
  Wird ein Tonsignal oder ein Dauerton gestartet, wird das Abspielen einer eventuell gerade ausgeführten anderen Audiodatei abgebrochen.
- Abspieldauer:
  - Tonsignal: Die Audiodatei bestimmt die Länge des Tonsignals.

- Dauerton: Der Aufruf der Funktionen **Dauerton Start** und **Dauerton Stopp** bestimmt die Länge des Tons.

- Es können mehrere Funktionen für unterschiedliche Tonsignale und Dauertöne angelegt und aufgerufen werden.
  Dazu müssen sie einzeln mit jeweils einem eigenen Button verknüpft werden.
- Werden mehrere Audiodateien in einem Skript gebündelt, wird nur die letzte Audiodatei abgespielt.
- Eine Umschaltung auf ein anderes Bild unterbricht das Abspielen einer Audiodatei nicht.

11.3.5 Variablen

In der HTML Web Engine werden diese Variablen angemeldet:

- Alle Variablen des jeweils aktuellen Bildes.
  Bei der Bildumschaltung werden Variablen des vorangegangenen Bildes abgemeldet und Variablen des neuen Bildes angemeldet.
- **Permanent überwachte Variablen**:
  Variablen, die bei Wertänderung einen Funktionsaufruf bewirken. Sie bleiben permanent angemeldet.
  Für sie gilt:
  - Die Variable hat mindestens einen Grenzwert.
  - Mindestens ein Grenzwert hat eine verknüpfte Funktion, die von der HTML Web Engine unterstützt wird.

> **Achtung**
>
> **Permanent überwachte Variablen** sind nicht ident mit Variablen, für die in zenon die Eigenschaft **Variable ständig lesen** aktiviert wurde.
>
> In zenon bewirkt diese Eigenschaft, dass die Variable am Treiber angemeldet wird und alle Änderungen der Steuerung immer an zenon gemeldet werden.
>
> In der HTML Web Engine wird automatisch entschieden, ob eine Variable ständig von der Runtime gelesen werden muss. Zum Beispiel, um bei einer Grenzwertverletzung automatisch ein Tonsignal abzuspielen.

VERHALTEN VON PERMANENT ÜBERWACHTE VARIABLEN

Um die Anzahl der Verbindungen für das Lesen von **Permanent überwachte Variablen** niedrig zu halten, verhalten sich Variablen wie folgt:

- Alle Variablen des Bildes werden von jeder einzelnen HTML Web Engine Session verknüpft.

- Alle **Permanent überwachte Variablen** werden nur einmalig verknüpft und alle HTML Web Engine Sessions nutzen dieselbe Verknüpfung. Dadurch wird gewährleistet, dass Änderungen von Werten der **Permanent überwachte Variablen** nicht für jede Session einzeln übertragen werden müssen.

Hinweise:

- **Permanent überwachte Variablen** werden unabhängig von den Variablen des aktuellen Bildes verknüpft.
  Dadurch können Variablen doppelt verknüpft werden. Dies hat zum aktuellen Zeitpunkt keine Auswirkung. Denn Änderungen von Werten der **Permanent überwachte Variablen** bewirken einen Funktionsaufruf nur bei Grenzwertverletzung. Dieser findet bei Änderungen von Werten der Bild-Variablen nicht statt.
- Das Umschalten zwischen verschiedenen Bildern bewirkt, dass die Variablen des vorhergehenden Bildes abgemeldet und jene des neuen Bildes verknüpft werden. **Permanent überwachte Variablen** werden dabei nicht angetastet.

INITIALE VERKNÜPFUNG

Die initiale Verknüpfung der **Permanent überwachte Variablen** erfolgt durch jenen Benutzer, der erstmalig eine HTML Web Engine-Session aufbaut. Sie bleibt so lange bestehen, bis alle HTML Web Engine Sessions, die diese Verknüpfung benutzen wieder abgebaut wurden. Das bedeutet: Später hinzugekommene HTML Web Engine Sessions können **Permanent überwachte Variablen** weiterhin nutzen, auch wenn der Benutzer, für dessen HTML Web Engine Session die Verknüpfung der **Permanent überwachte Variablen** durchgeführt wurde, bereits abgemeldet ist.

VERBINDUNGSABBRUCH UND VERBINDUNGSWIEDERHERSTELLUNG:

Im Fall eines Verbindungsabbruchs werden alle Variablen in allen HTML Web Engine Sessions abgemeldet.

Wurde die Verbindung wieder hergestellt, stehen **Permanent überwachte Variablen** erst wieder zur Verfügung, wenn sich die Benutzer erneut anmelden.

FUNKTIONSAUFRUF ÜBER GRENZWERTVERLETZUNG

Die HTML Web Engine überprüft für **Permanent überwachte Variablen**, ob eine Grenzwertverletzung für diese Variable aufgetreten ist. Wird eine Grenzwertverletzung erkannt, dann wird die beim Grenzwert hinterlegte Funktion ausgeführt.

Dabei gilt:

- Die Funktion wird ausgeführt, unabhängig davon, ob die Variable im Bild verfügbar ist.
- Die Funktion wird für alle angemeldeten Benutzer ausgeführt.
- Die Funktion muss von der HTML Web Engine unterstützt werden.

- Haben zwei Grenzwerte den selben Wert, dann ist nicht festgelegt, welche ausgeführt wird. Die Reihenfolge muss beim Web Engine Compiler berücksichtigt werden.

Definition Grenzwertverletzung für HTML Web Engine:

- Die Eigenschaft **Grenzwert aktiv** muss für die Variable aktiv sein.
- Wenn für die Eigenschaft **Minimum/Maximum** *Maximum* konfiguriert wurde, gilt:
  - Die Grenzwerte werden absteigend sortiert.
  - Der erste Grenzwert, der kleiner/gleich dem aktuellen Variablenwert ist wird angewendet.
  - Der Wert für **Grenzwert** muss größer als der bisherige Variablen-Wert sein.
- Wenn für die Eigenschaft **Minimum/Maximum** *Minimum* konfiguriert wurde, gilt:
  - Die Grenzwerte werden aufsteigend sortiert.
  - Der erste Grenzwert, der größer/gleich dem aktuellen Variablenwert ist wird angewendet.
  - Der Wert für **Grenzwert** muss kleiner als der bisherige Variablen-Wert sein.

12 Projekt für Web kompilieren

Mit dem **Web Engine Compiler** werden aus einem zenon Projekt heraus jene Daten generiert, die die **HTML Web Engine** benötigt, um die HTML5 Inhalte für den zenon Web Client zur Verfügung zu stellen. Beim Übersetzungsvorgang dieser Projektdaten überprüft der **Web Engine Compiler** die Projektinhalte und gibt Auskunft über nicht unterstützte Funktionen oder Eigenschaften. Als Resultat des Übersetzungsvorgangs entsteht eine Datei, die der **Web Engine** zur Verfügung gestellt wird.

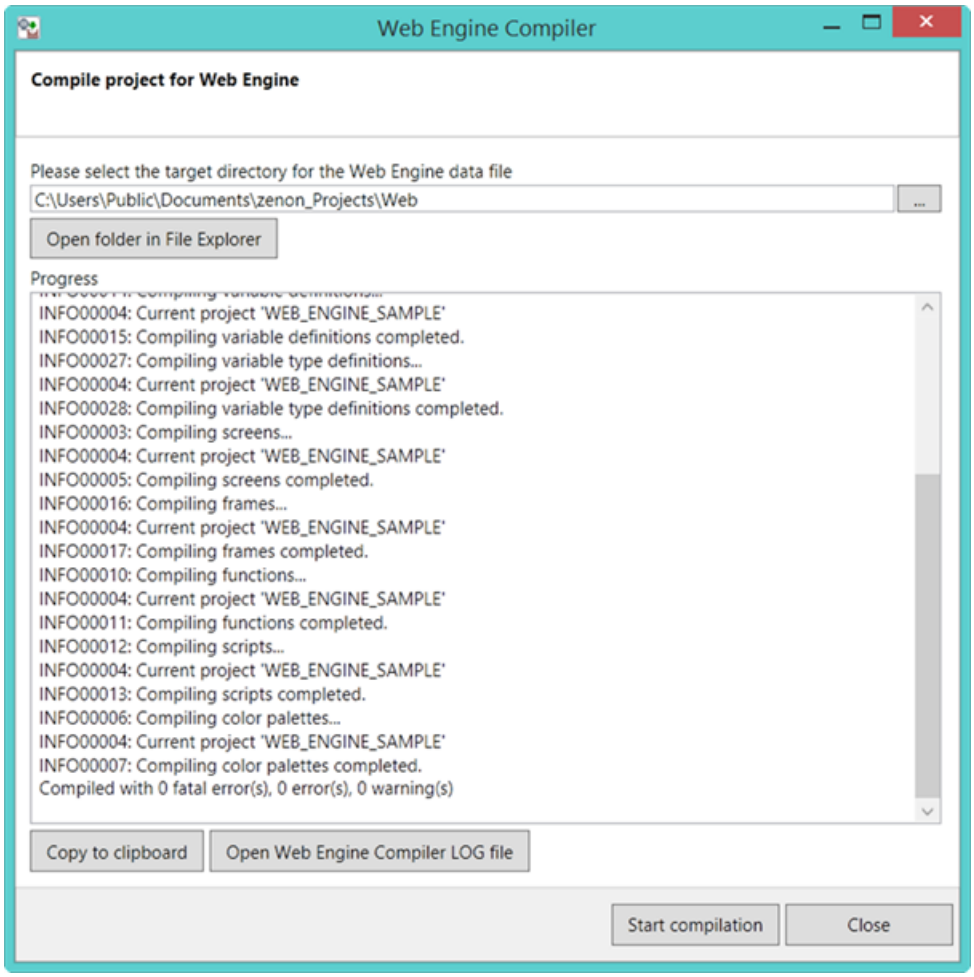
Um den **Web Engine Compiler** zu öffnen:

1. Klicken Sie in der Menüleiste des Editors auf **Extras**.
2. Klicken Sie auf **Projekt für Web kompilieren...**

   Der Dialog zum Kompilieren des Projekts wird geöffnet.
   Hinweis: Dieser steht nur in Englisch zur Verfügung.

> **Achtung**
>
> Wenn der **Web Engine Compiler** mehrfach für unterschiedliche zenon Projekte ausgeführt wird, werden auch mehrere Web Engine Projektdateien im Ausgabeordner abgelegt. In diesem Fall benutzt die HTML Web Engine immer nur die im Alphabet erstgereihte Projektdatei.

DIALOG HTML WEB ENGINE COMPILER



Option	Beschreibung
Please select the target directory for the Web Engine data file	Wählen Sie hier einen Ordner zum Speichern der **Web Engine** Datei aus. Klick auf Schaltfläche **...** öffnet Dialog zur Auswahl eines Ordners.
Open folder in File Explorer	Öffnet den ausgewählten Ordner im Windows Explorer.
Progress	Zeigt beim Kompilieren Warnungen, Fehlermeldungen und Informationen an.
Copy to clipboard	Kopiert den Inhalt des Ausgabefensters **Progress** in die Zwischenablage.
Open Web Engine Compiler LOG file	Öffnet die **Web Engine Compiler** LOG-Datei. Diese enthält die Meldungen aus dem Ausgabefenster **Progress**.

Option	Beschreibung
Start compilation	Startet den Kompiliervorgang.
Close	Schließt den **Web Engine Compiler.**

13 Ablauf einer HTML Web Engine Sitzung

Nach erfolgreichem Kompilieren der Projektdaten, steht die HTML Visualisierung zur Verfügung.
Um die Webseite aufzurufen:

1. Öffnen Sie einen HTML5 kompatiblen Web Browser.
   Die Liste der empfohlenen Web Browser, finden Sie im Kapitel Systemvoraussetzungen (auf Seite 12).
2. Geben Sie in die Adressleiste des Web Browsers die Webseiten-URL für die HTML5 Visualisierung ein: **https://Serveradresse/Applikationsname**.

   Die Bereitstellung der HTML5 Inhalte erfolgt automatisch. Dabei wird für jeden zenon Web Client eine eigene Sitzung erstellt und verwaltet. Die Laufzeitdaten der zenon Runtime stehen zur Verfügung, sobald Sie als Benutzer erfolgreich authentifiziert wurden. Mehr Details dazu lesen Sie im Kapitel **Client Authentifizierung für eine Verbindung zur Runtime** (auf Seite 26).
3. Sobald Sie die Webseite verlassen, werden die HTML Web Engine Sitzung und die Verbindung zur zenon Runtime automatisch abgebaut.

Hinweis: Das Verlassen der Webseite erfolgt unter anderem durch Schließen des Web Browsers, Schließen der Registerkarte, Aktualisierung der Ansicht oder erneute Eingabe der URL.

WEB ENGINE SITZUNG NEU STARTEN

Wenn ein Neustart der Web Engine Sitzung nötig ist, wird dies über einen Dialog angezeigt. Der Neustart kann automatisch oder manuell erfolgen. Per Default startet der Dialog die Sitzung nach 30 Sekunden selbständig neu. Die verbleibende Zeit bis zum Neustart wird dabei angezeigt.
Diese Zeitspanne kann individuell konfiguriert werden.

Um die Zeitspanne bis zum Neustart zu konfigurieren:

1. Öffnen Sie die Datei **appsettings.json** mit einem Texteditor.
   Diese finden in dem Ordner, in dem die Web Engine publiziert wurde.
   Default: **C:/inetpub/wwwroot/<Name Web Engine>**
2. Navigieren Sie zum Eintrag **AutomaticSessionRestartDelay**.
3. Tragen Sie die gewünschte Verzögerung in Sekunden ein.

- Minimum: *0*
  Bei negativen Werten, *0* oder fehlendem Eintrag erfolgt kein automatischer Neustart. Der Neustart muss dann manuell über Klick auf die Schaltfläche **Neustart** durchgeführt werden.
- Maximum: *3600* (= *1 h*)

Default: *30*

14 Systemdiagnose und Fehlersuche

Sollte es beim Systemstart oder Betrieb zu Problemen kommen, werden am HTML Web Client Fehlermeldungen angezeigt, die Auskunft über die mögliche Problemursache geben. Die Protokollierungsstufe der Fehlermeldungen hängt davon ab, wie die HTML5 Webseite aufgerufen wird:

- Lokaler Web Browser: Sie erhalten detaillierte Meldungen.
  Dazu muss der Web Browser auf dem selben Rechner (mit der gleichen IP Adresse) laufen, auf dem auch die Web Engine betrieben wird.
- Remote Browser: Allgemeine Meldungen.
  Das gilt bei einem Aufruf von einem anderen Gerät und/oder einer anderen IP-Adresse.

CHECKLISTEN

FÜR DEN FEHLERFREIEN SYSTEMBETRIEB

Zur allgemeinen Prüfung der Systemkonfiguration werden folgende Überprüfungen empfohlen:

- HTML Web Engine wurde am IIS installiert.
  - Das Web Engine Deployment wurde fehlerfrei durchgeführt.
  - Der Web Server ist in Betrieb.
- Visualisierungsdaten wurden mit dem **Web Engine Compiler** erzeugt.
  - Es kommt zu keinen Fehlern beim Kompiliervorgang.
  - Die resultierenden Daten des **Web Engine Compiler** stehen für den Zugriff durch die Web Engine bereit.
    Hinweis: Das Auftreten von Warnungen beeinflusst die Lauffähigkeit der HTML Web Engine nicht grundsätzlich. Es kann je nach Art der Warnung jedoch zu spezifischen Einschränkungen der projektierten Funktionalität kommen.

- Die Versionen von **Web Engine Compiler** und HTML Web Engine sind identisch.
  Hinweis: Die vom **Web Engine Compiler** erzeugten Daten können nur von der HTML Web Engine (Web Applikation) mit gleicher Versionsnummer richtig interpretiert werden.

FÜR DIE ÜBERTRAGUNG VON PROZESSDATEN

Beachten Sie bei der Übertragung von Prozessdaten:

- zenon Runtime und der konfigurierte Connector wurden gestartet. Der Connector ist über das Netzwerk erreichbar.
  Hinweis: Die Prozesse für zenon Runtime und **SCADA Runtime Connector** müssen im gleichen Benutzerkontext laufen.
- Die Benutzer, mit denen die Authentifizierung erfolgt, müssen in der zenon Runtime verfügbar sein.