



# Handbuch

## MLXML Business Integration Connector

© 2001-2026 by delight software gmbh

# MLXML Business Integration Connector

## 2025

Basierend auf delight base 13.x

© 2001-2026 by *delight software gmbh*



# **MLXML Business Integration Connector 2025**

© 2001-2026 by delight software gmbh

Es gelten die Allgemeinen Geschäftsbedingungen und die Allgemeinen Lizenzvereinbarungen der  
delight software gmbh

# Inhaltsverzeichnis

<b>Kapitel 1 - BI Connector</b>	<b>2</b>
<b>Kapitel 2 - Konfiguration</b>	<b>4</b>
2.1 Lokaler Benutzer .....	5
2.1.1 SSL Zertifikat aktivieren .....	6
2.2 Server .....	7
2.2.1 Öffentliche REST API .....	8
2.2.2 SSL Zertifikat aktivieren .....	9
<b>Kapitel 3 - BI-Konten</b>	<b>12</b>
3.1 Externer Kundenzugriff .....	13
3.1.1 Berechtigung .....	14
3.2 Konfiguration .....	15
3.2.1 Kontenübersicht .....	15
3.2.1.1 Erlaubte API Methoden .....	16
3.2.1.2 Der "publicdatauser" .....	16
<b>Kapitel 4 - Verbindung</b>	<b>18</b>
4.1 WDSL .....	19

**BI Connector**

**1**

# 1 BI Connector

Der Business Integration Connector stellt eine moderne JSON REST-API sowie einen legacy SOAP-Webservice, für den Zugriff von externen Programmen, zur Verfügung.

In der Netzwerk-Version des Produktes enthält jeder Client einen separaten, lokalen Connector. Damit ist es möglich, externe Anwendungen direkt mit dem jeweiligen Client-Programm zu verbinden. Ein Anwendungsbeispiel ist die Archivierung von E-Mail aus einem externen E-Mailprogramm. Ab delight v2025 ist der lokale Connector für externe E-Mail-Addons (Outlook, Thunderbird, Office) nicht mehr erforderlich. Die aktuellen AddOns können (und sollten) die [zentrale REST-API des Servers](#) verwenden.

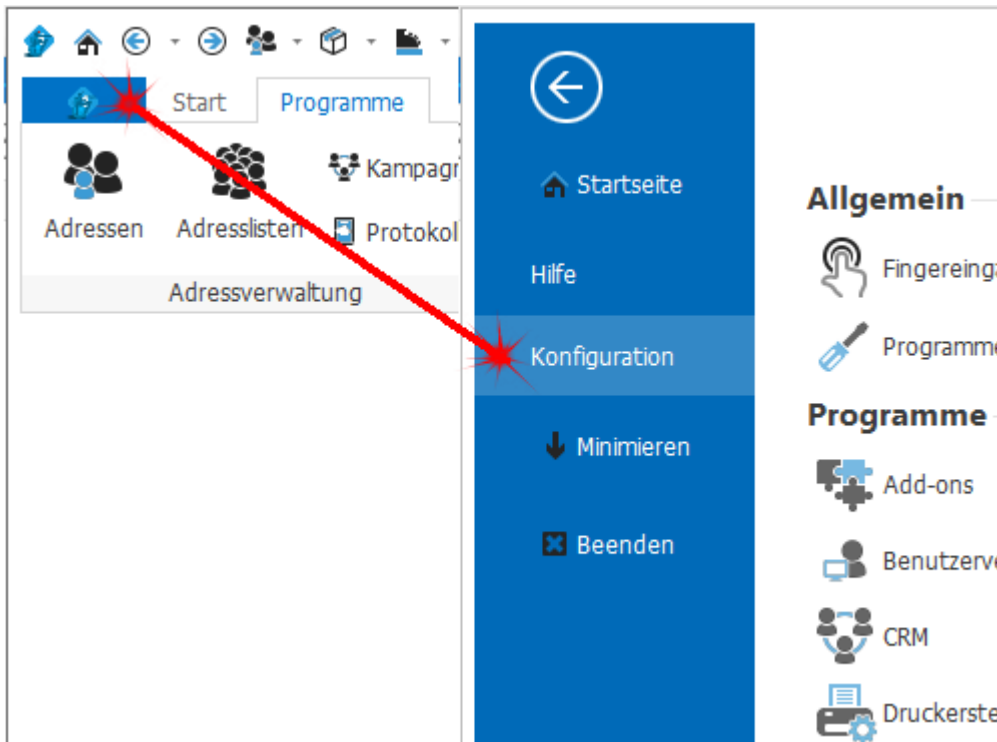
**Konfiguration**

**2**

## 2 Konfiguration

Die Einstellungen werden in der Backstage-Ansicht unter *Konfiguration* => *BI Connector* aufgerufen.

In der Business Integration Connector Konfiguration kann der TCP/IP Port für die REST-API sowie den legacy SOAP-Server konfiguriert werden.



### 2.1 Lokaler Benutzer

Im Reiter *Lokaler Benutzer* kann der TCP/IP Port für den aktuellen Benutzer definiert werden. Diese Einstellung wird Benutzer- und Computerabhängig gespeichert. In der Netzwerk-Version kann der Port alternativ auch direkt in der Benutzerverwaltung auf dem jeweiligen Benutzer definiert werden.



Lokaler Benutzer
Server

BI-SOAP-SCHNITTSTELLE LOKALER BENUTZER

SOAP TCP/IP Port:  
8090
Automatisch bestimmen

Lokaler BI-Connector URL für Benutzer:  
http://127.0.0.1:8090  
http://::1:8090

BI-REST-SCHNITTSTELLE LOKALER BENUTZER

REST TCP/IP Port:  
9090
Automatisch bestimmen

☐ SSL verwenden
HTTP-Modus: Socket (fallback)

Lokaler BI-Connector URL für Benutzer:  
http://DELIGHTDOC:9090

Bevor SSL aktiviert werden kann, müssen Sie das Server-Zertifikat importieren und für HTTP.SYS auf dem PC aktivieren! SSL funktioniert nur mit http.sys, der Socket-Fallback unterstützt kein SSL!  
[Zertifikat mit Hash aktivieren](#)  
HTTP.sys neu konfigurieren  
Lokales SSL-Zertifikat für REST-API konfigurieren

Wenn Sie mehrere delight Programme parallel auf dem selben Computer (oder auf einem Terminal-Server) betreiben, muss in jedem Programm ein individueller Port definiert werden. Über diesen Port können die externen Programme (zB. Add-Ins in externen E-Mailprogrammen) auf das delight Programm zugreifen.

Wir bei Port der Wert 0 eingetragen, wird die Schnittstelle deaktiviert.

### 2.1.1 SSL Zertifikat aktivieren

Dieses Kapitel richtet sich an **Systemtechniker und Experten**

Für die lokale Benutzer REST-API muss auch im produktiven Betrieb nicht unbedingt ein SSL-Zertifikat aufgeschaltet werden. In der Regel wird der Benutzer REST API Service "nur" von localhost verwendet.

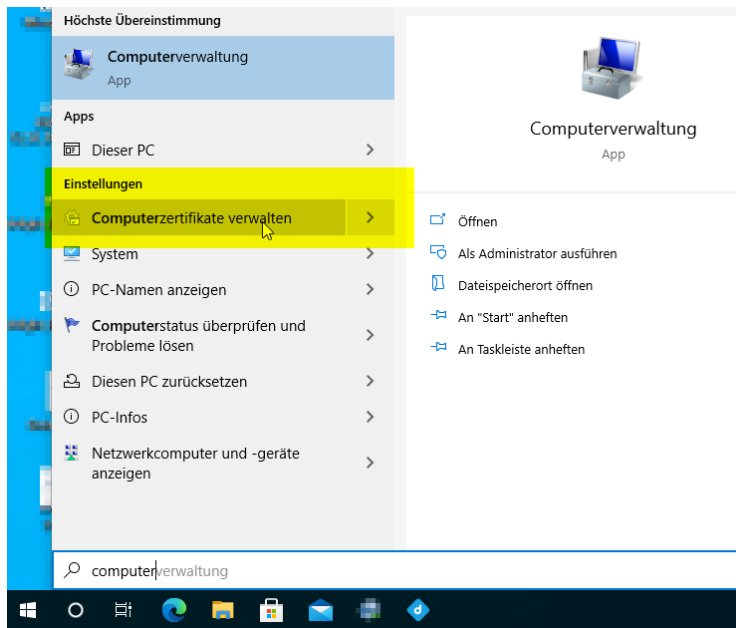
Bei Bedarf können auch selbst signierte Zertifikate oder kostenlose Zertifikate von Let's Encrypt aufgeschaltet werden.

HTTPS wird nur im sogenannten HTTP.sys Kernel-Modus unterstützt. Bei Microsoft's HTTP.sys können Zertifikate über die Befehlszeile "netsh http add sslcert" installiert werden. Das Zertifikat muss dazu im "Computer lokalen" Windows Zertifikatsspeicher installiert sein. Um den Prozess etwas zu vereinfachen, kann ein im Zertifikatsspeicher installiertes Zertifikat in delight per Klick ausgewählt und aktiviert werden.

#### Zertifikat per Klick installieren

Für diese Installation werden Administratorrechte benötigt.

1. Das gewünschte Zertifikat muss im Windows Zertifikatsspeicher installiert sein und Sie müssen im Besitz des PrivateKey's sein.



2. Starten Sie die ClientApp (ML2Client.exe) auf dem Computer des Benutzers.
3. Öffnen Sie *Konfiguration* => *BI Connector* > *Lokaler Benutzer*
4. Konfigurieren Sie zuerst den REST TCP/IP Port. Das müssen Sie unbedingt VOR der Aktivierung des Zertifikats tun, da bei HTTP.sys ein Zertifikat immer auf einen bestimmten TCP/IP Port installiert wird. Wenn Sie den Port ändern, müssen Sie das SSL-Zertifikat erneut aktivieren.
5. Setzen Sie den Hacken bei *SSL verwenden*  
 REST TCP/IP Port:  
 9090  ☐ *SSL verwenden*
6. Klicken Sie auf *Lokales SSL-Zertifikate für REST-API konfigurieren*  
 Lokales SSL-Zertifikat für REST-API konfigurieren  
 Wenn eines oder mehrere verwendbare Zertifikat installiert sind, kommt eine Auswahl. Wählen Sie das gewünscht Zertifikat und bestätigen Sie mit *OK*.
7. Nach dem Klicken wird eine zweite Instanz der Anwendung im Konfigurationsmodus gestartet. Allenfalls kommt eine Aufforderung der Windows-Elevation die Sie bestätigen müssen.

## 2.2 Server

Dieses Kapitel richtet sich an **Systemtechniker und Experten**

Im Reiter *Server* befinden sich die zentralen Server-Einstellungen zur REST-API sowie zur legacy SOAP-Schnittstelle.

Die hier definierten TCP/IP Ports stehen auf den Interfaces des zentralen delight Windows-Dienst zur Verfügung.

Der REST-API Port wird z. B. für delight App & Portal oder die E-Mail-Addons benötigt. Für den produktiven Einsatz wird das aktivieren der [SSL-Verschlüsselung](#) dringend empfohlen.

Der legacy SOAP-Port wird nur in Ausnahmefällen benötigt.

Lokaler Benutzer
Server

BI-SCHNITTSTELLE FÜR MULTI-USER SERVER

Benutzername:
Passwort:

SOAP TCP/IP Port:
0

REST TCP/IP Port:
9092
☒ SSL verwenden

Access-Control-Allow-Origin:
\*

Bevor SSL aktiviert werden kann, müssen Sie das Server-Zertifikat für HTTP.SYS auf dem Server importieren! SSL funktioniert nur mit http.sys, der Socket-Fallback unterstützt kein SSL!

WICHTIG: Installation des Zertifikats muss LOKAL auf dem Server (delight Window Dienst) durchgeführt werden. Experten-Installation auch manuell über "netsh http add sslcert" möglich.

[Zertifikat mit Hash aktivieren](#)

HTTP-Modus abfragen

HTTP.sys neu konfigurieren

Lokales SSL-Zertifikat für REST-API konfigurieren

Dienst neu starten

Wir bei Port der Wert 0 eingetragen, wird die Schnittstelle deaktiviert.

### Benutzername und Passwort

Zugangsdaten für den Super-Admin Zugang. Dieses Angaben sollten Sie in produktiven Umgebungen leer lassen (=deaktiviert). Verwenden Sie stattdessen das Menü BI-Konten.

### Wichtig:

Diese Einstellung sollte nur für Administratoren sichtbar sein.

## 2.2.1 Öffentliche REST API

Dieses Kapitel richtet sich an **Systemtechniker und Experten**

In vielen Fällen, z. B. auch für den Betrieb der delight App oder des delight Kunden-Portals, wird eine öffentliche, extern im Internet verfügbare REST-API benötigt. Ab v2025 wird diese zentrale REST-API auch für die E-Mail-Addons (Outlook, Thunderbird, Office...) benötigt.

### Beachten Sie dazu bitte folgenden Punkte zum Thema Datenschutz und Sicherheit:

- Alle Benutzerkonten müssen mit einem starken Passwort geschützt sein. Dies gilt insbesondere auch für die normalen Benutzerkonten der Mitarbeiter.
- Prüfen Sie auch allfällige "admin" Benutzer. Wir stellen immer wieder fest, dass das Standard-Passwort nicht geändert wurde oder gar leer ist.
- Die REST-API muss unbedingt mit einer SSL-Verschlüsselung konfiguriert werden.
- Sollten Sie API-User mit der Rolle "Systembenutzer" benötigen, schützen Sie diese ebenfalls mit sehr starken Passwörtern und definieren Sie die [Erlaubten API Methoden](#).
- Deaktivieren Sie den [Super-Admin Zugang](#).
- Schränken Sie den Kreis der Mitarbeiter, welche [externe Kundenzugriffe bearbeiten können](#), auf eine Minimum ein.

**Überlegungen und Empfehlungen zur Installation:**

- Wenn möglich sollte Port 443 verwendet werden, da gewisse Firmen andere Ports blockieren. Ist der HTTP/SSL-Port 443 bereits durch eine andere Anwendung besetzt, verwenden Sie für die API einen hohen TCP/IP Port. z. B. unseren Standard-Port Vorschlag 9090.
- Verwenden Sie ein offizielles, verifizierbares SSL-Zertifikat. Keine self-signed Zertifikate. Unsere Empfehlung: Verwenden Sie kostenlose Let's Encrypt Zertifikate.
- Sollten der CRM-Server-Standort keine fix IP-Adresse haben, ist auch ein DynDNS möglich. Wir würden in diesem Fall neben dem DynDNS-Eintrag noch einen weiteren DNS-Eintrag auf Ihrer eigenen Domäne empfehlen. Bsp. Wenn Ihre Website [www.mycompany.ch](http://www.mycompany.ch) wäre und der DynDNS des CRM-Server Standorts "mycompany.dyndns.org" wäre, sollten Sie zusätzlich folgenden CNAME-Eintrag erstellen: [delightapi.mycompany.ch](http://delightapi.mycompany.ch) typ CNAME auf mycompany.dyndns.org  
Der API-Endpunkt (der z. B. in die delight App&Portal-Konfiguration übernommen werden muss) wäre in diesem Bsp. dann <https://delightapi.mycompany.ch:9090>

Der Vorteil dieser Lösung: Sollte der CRM-Server irgend wann einmal umziehen (z. B. in eine Cloud oder ein eigenes RZ, anderer DynDNS-Provider weil Hardware wechselt) muss nur der DNS-Eintrag [delightapi.mycompany.ch](http://delightapi.mycompany.ch) angepasst werden. Eine Neukonfiguration aller externen System welche die API verwenden ist nicht erforderlich.

- Installieren Sie das SSL-Zertifikat wie [hier](#) beschrieben für HTTP.sys, verwenden Sie kein SSL offloading auf der Firewall.
- Die delight REST-API basiert auf Microsoft's Kernel HTTP.sys. Nach unseren Tests sind damit auf einfachen Servern bereits sehr hohe Durchsätze bis zu 35'000 Requests / Sekunden und mehr möglich. Der Flaschenhals liegt hier oftmals bei der Datenbank. d. h. für Optimierungen ist dort der erste Ansatzpunkt.
- Im delight App&Portal Hosting ist zusätzlich eine API-Cache Projekt integriert. Sollten Sie ausgewählte Daten öffentlich über eine API zur Verfügung stellen wollen, sollte dies über den API-Cache konfiguriert werden. Der API-Cache ist nicht Teil dieser Dokumentation und wird nur im Rahmen unseres Entwickler-Supports zur Verfügung gestellt bzw. supported.

## 2.2.2 SSL Zertifikat aktivieren

Dieses Kapitel richtet sich an **Systemtechniker und Experten**

Für die zentrale Server REST-API sollte im produktiven Betrieb unbedingt ein SSL-Zertifikat aufgeschaltet werden.

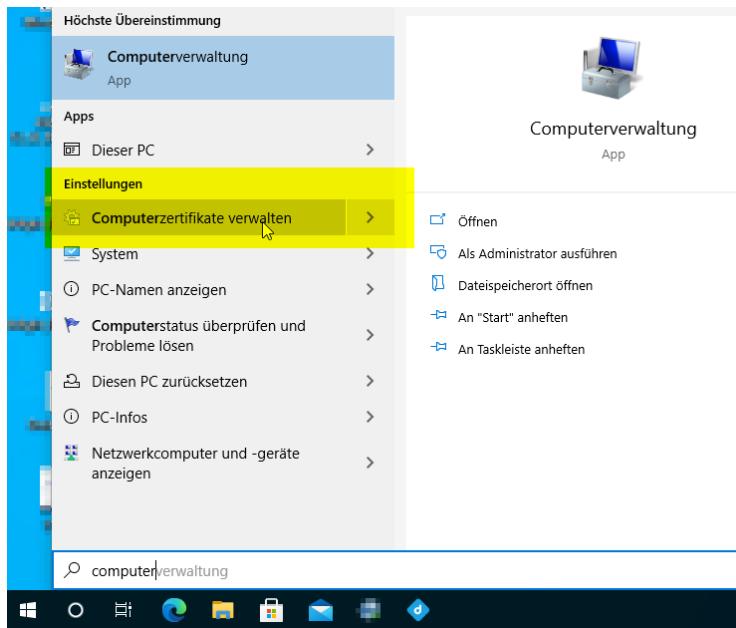
Externe App & Portal Anwendungen können aus Sicherheitsgründen nicht ohne SSL betrieben werden und benötigen ein offizielles SSL-Zertifikat auf der REST-API. Selbst signierte Zertifikate sind für App und Portal nicht geeignet. Kostenlose Zertifikate von Let's Encrypt funktionieren problemlos.

HTTPS wird nur im sogenannten HTTP.sys Kernel-Modus unterstützt. Bei Microsoft's HTTP.sys können Zertifikate über die Befehlszeile "netsh http add sslcert" installiert werden. Das Zertifikat muss dazu im "Computer lokalen" Windows Zertifikatsspeicher installiert sein. Um den Prozess etwas zu vereinfachen, kann ein im Zertifikatsspeicher installiertes Zertifikat in delight per Klick ausgewählt und aktiviert werden.

**Zertifikat per Klick installieren**

Für diese Installation werden Administratorrechte benötigt.

1. Das gewünschte Zertifikat muss im Windows Zertifikatsspeicher installiert sein und Sie müssen im Besitz des PrivateKey's sein.



2. Starten Sie die ClientApp (ML2Client.exe) direkt auf dem Server auf dem der delight Window Service installiert ist und läuft.
3. Öffnen Sie *Konfiguration => BI Connector > Server*
4. Konfigurieren Sie zuerst den REST TCP/IP Port. Das müssen Sie unbedingt VOR der Aktivierung des Zertifikats tun, da bei HTTP.sys ein Zertifikat immer auf einen bestimmten TCP/IP Port installiert wird. Wenn Sie den Port ändern, müssen Sie das SSL-Zertifikat erneut aktivieren.
5. Setzen Sie den Hacken bei *SSL verwenden*

REST TCP/IP Port:  
 ☒ SSL verwenden

6. Klicken Sie auf *Lokales SSL-Zertifikate für REST-API konfigurieren*

Lokales SSL-Zertifikat für REST-API konfigurieren

Wenn eines oder mehrere verwendbare Zertifikat installiert sind, kommt eine Auswahl. Wählen Sie das gewünscht Zertifikat und bestätigen Sie mit *OK*.

7. Nach dem Klicken wird eine zweite Instanz der Anwendung im Konfigurationsmodus gestartet. Allenfalls kommt eine Aufforderung der Windows-Elevation die Sie bestätigen müssen.
8. Warten Sie, bis die Konfiguration durchgeführt wurden und klicken Sie anschliessend auf *HTTP-Modus abfragen*. Wenn darauf die Meldung kommt, dass der Server im HTTP.sys (fast) Modus arbeite, dürfte es geklappt haben.
9. Die Funktion der API kann mit einem simplen Aufruf in jedem gängigen Webbrowser geprüft werden.

Bsp: <https://mydelight.mysite.ch:9090/services/system/GetVersion>

(natürlich muss der DNS-Name und der Port durch die korrekten Werte Ihrer Installation angepasst werden)

Liefert dieser Aufruf ein eine Error-Response im JSON Format, funktioniert die API:

```
{
  "errorCode": 401,
  "errorText": "#0 Session not found or expired!"
}
```



**BI-Konten**

**3**

### 3.1 Externer Kundenzugriff

Susi Schnell

Details

Journal

Termine

Ablage

Kampagnen

Bilder

Notizen

Versandvorgaben

Zeit- und Leistungserfassung

Passwörter

ADRESSE

Firmenname:

Anrede:

Vorname:

Name:

Adresse:

PLZ:

Ort:

Land:

Frau

Susi

Schnell

Echoweg 15

1234

Vessy

Schweiz

KONTAKT

Telefon-Nr privat:

Telefon-Nr geschäftlich:

Telefon-Nr weitere:

Mobil privat:

Mobil geschäftlich:

Hauptnummer:

Fax privat:

Fax geschäftlich:

Email privat:

Email geschäftlich:

Email weitere:

Internet privat:

Internet weitere:

Pager privat:

Pager geschäftlich:

MSN-Email:

ICQ Nummer:

Skype privat:

Skype geschäftlich:

066 345 34 12

079 122 12 12

su.schnell@gmx.com

DETAILS

Nationalität:

Korrespondenzsprache:

Geburtsdatum:

Alter [Jahre]:

Zivilstand:

Interessen:

Eigenschaften:

Schweiz

Französisch

0

ledig

Sport, Freizeit

Geburtstagskarte

WEITERES

Adress-Nr.:

Kundenbetreuer:

Status:

Tätigkeitsstatus:

Personen Nummer:

17

Gordola Peter

aktiv

BANKVERBINDUNGEN

Bank:

Konto-Nr.:

IBAN-Nummer:

ZUORDNUNGEN

Auswahl...

Kunden

Kunden A

Lieferanten

BEWERTUNGEN

Start

||

AUFTRAGE

Nummer

Titel

Referenz

Verkaufsprozess

<keine Daten zum anzeigen>

AUFGABEN

Erlidgt

Aufgabenart

Aufgabenstatus

Zuständiges Team

<Keine Aufgaben>

EXTERNER KUNDENZUGRIFF

Benutzername

Passwort

Gültig ab

Gültig bis

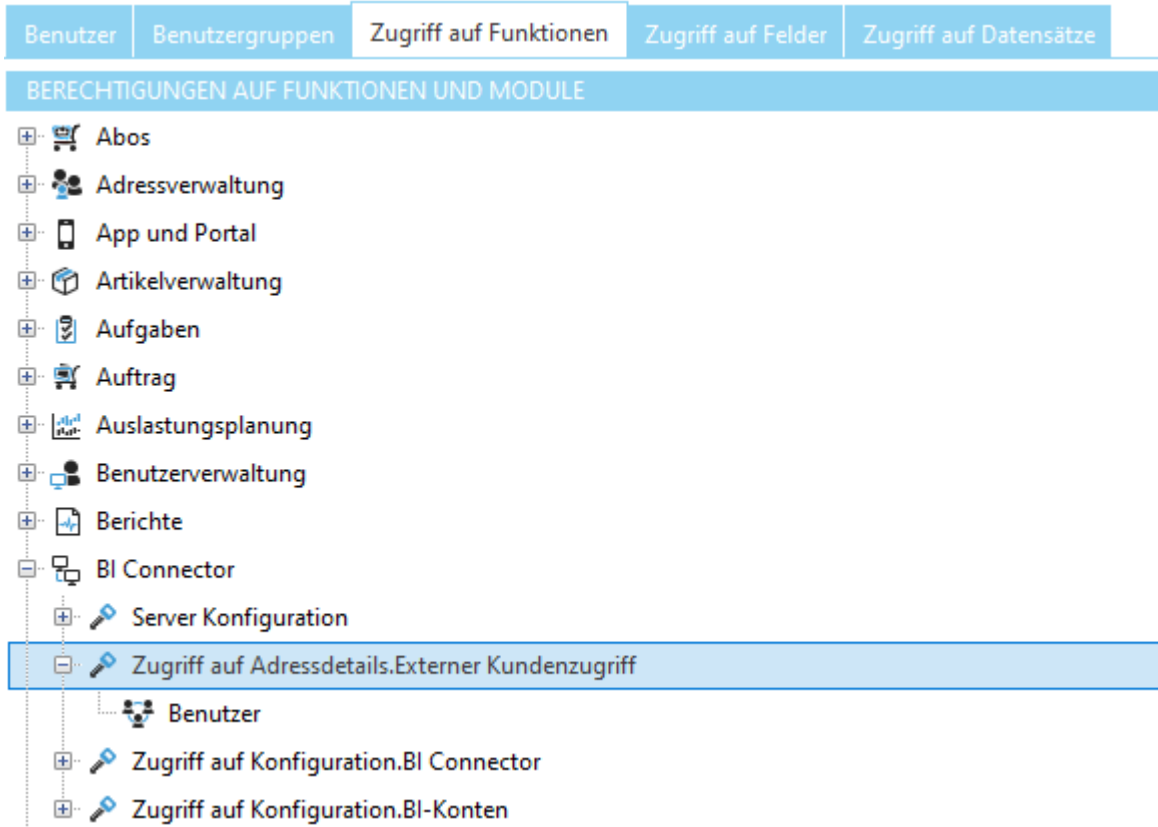
lara

\*\*\*\*



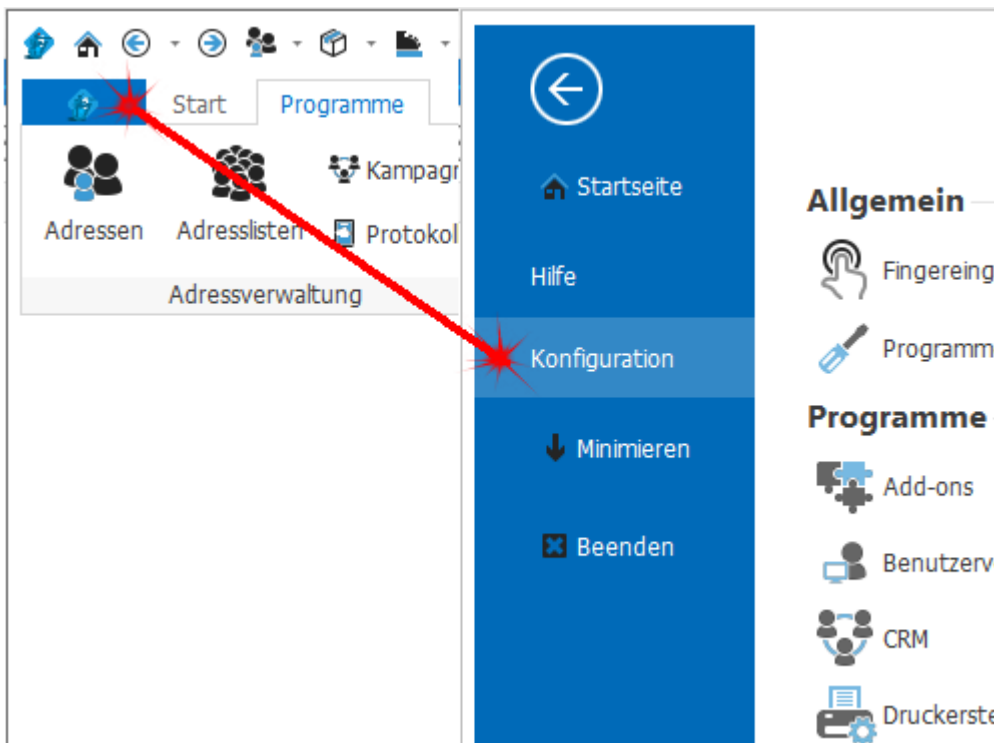
### 3.1.1 Berechtigung

Ob der Block *Externer Kundenzugriff* für bestimmte Benutzer verfügbar ist, kann über die Berechtigung *BI Connector* => *Zugriff auf Adressdetails.Externer Kundenzugriff* konfiguriert werden.



## 3.2 Konfiguration

Die Einstellungen werden in der Backstage-Ansicht unter *Konfiguration* => *BI-Konten* aufgerufen.



### 3.2.1 Kontenübersicht

In der BI-Kontenübersicht werden alle Zugriffskonten der REST-API aufgelistet und verwaltet.

BI-Konten						
Benutzername	Passwort	Rolle	Person	Erlaubte API Methoden	Gültig ab	Gültig bis
lara	****	Kunde	Susi Schnell			
h.muster	*****	Kunde	Firma Rex Hans Muster			31.12.2023
publicdatauser	****	Systemvollzugriff		database/table/QueryRecor		
apicacheuser	*****	Systemvollzugriff		database/table/QueryRecor		

#### Rolle

Die Rolle des Kontos definiert, welche Berechtigungen und Zugriffe über die API möglich sind. Für Kundenzugriffe wird die Rolle "Kunde" verwendet. Diese Konten erhalten automatisch nur Zugriff auf die Kundendaten die unter "Person" angegeben sind. Die Rolle Systemvollzugriff erhält Vollzugriff. Entsprechend sollten Sie solche Konten nicht leichtfertig erstellen und in der Spalte "Erlaubte API Methoden" Einschränkungen auf das nötigste vornehmen.

Die Rolle "System Benutzer" kann nicht ausgewählt werden, da für normale Programm-Benutzer an dieser Stelle keine Konten erfasst werden. Normale System-Benutzer (Mitarbeiter) erhalten automatisch API-Zugriff über ihren Benutzer der normalen Benutzer-Verwaltung. Damit erhalten alle Mitarbeiter automatisch Zugriff auf die delight App.

#### Person

Verknüpfung auf die Person. i. d. R. Kunde der Rolle.

#### Erlaubte API Methoden

siehe [Erlaubte API Methoden](#)

### 3.2.1.1 Erlaubte API Methoden

---

Dieses Kapitel richtet sich an **Systemtechniker und Experten**

Im Feld *Erlaubte API Methoden* kann der Zugriff des Kontos auf bestimmte API-Methoden eingeschränkt und zusätzlich parametrisiert werden.

Wird bei der Rolle "Kunde" nicht benötigt. Bei Rolle "Systemvollzugriff" empfohlen.

Wenn dieses Feld leer ist, sind alle API-Methoden erlaubt (=Vollzugriff). Sobald Einschränkungen eingetragen werden, kann nur noch auf die eingetragenen Methoden zugegriffen werden.

Auf jeder Zeile kann eine API-Methode definiert und mit einem JSON-Wert parametrisiert werden:

```
die/api/methode={json-parameter}
```

Die genaue Funktionsweise ist nicht Teil dieser Dokumentation und wird nur im Rahmen unseres Entwickler-Supports zur Verfügung gestellt bzw. supported.

**siehe auch**

[Publicdata-User](#).

### 3.2.1.2 Der "publicdatauser"

---

Dieses Kapitel richtet sich an **Systemtechniker und Experten**

Für gewisse externe Anwendungen wird ein öffentlicher Benutzer benötigt. Über diesen Benutzer müssen öffentlich verfügbare Daten abgefragt werden können. Da die delight REST API grundsätzlich keine Daten ohne spezifischen Benutzer liefert, muss für solche Fälle ein eigener Benutzer erstellt werden. Diesen Benutzer wird Publicdata-User genannt und sollte nur dann erstellen, wenn er auch tatsächlich benötigt wird.

Aktuell wird dieser Benutzer im delight Kunden-Portal immer dort benötigt, wo Aktionen ohne Login möglich sind. Aktuell ist dies nur im Zusammenhang mit dem delight Support-Ticketsystem der Fall.

Der sogenannte "publicdatauser" muss bei Bedarf wie folgt konfiguriert werden:

BI-Konto bearbeiten

Start

Abbrechen Speichern Änderungen verwerfen

Dialog

Benutzername: publicdatauser

Passwort: \*\*\*\*

Rolle: Systemvollzugriff

Person:

Erlaubte API Methoden: database/table/QueryRecords={"SchemaName": {"grant": ["@table:ticketssystem\_ticketphases"]}} ticketsystem/PublicDataService/QueryTicketData={} ticketsystem/PublicDataService/AddTicket={} ticketsystem/PublicDataService/ExecuteTicketFunction={}

Gültig ab:

Gültig bis:

**Benutzername:** publicdatauser

**Passwort:** 1234

**Erlaubte API Methoden:**

```
database/table/QueryRecords={"SchemaName": {"grant": ["@table:ticketssystem_ticketphases"]}}
ticketsystem/PublicDataService/QueryTicketData={}
ticketsystem/PublicDataService/AddTicket={}
ticketsystem/PublicDataService/ExecuteTicketFunction={}
```

**Verbindung**

**4**

## 4 Verbindung

Damit externe Programme eine Verbindung herstellen können, müssen Sie im externen Programm (oder Add-In) den Verbindungs-URL eintragen. Der Verbindungs-URL wird in der [Konfiguration](#) angezeigt.

### **Authentifizierung:**

**Netzwerk-Version:** Benutzername und Passwort zum verbinden sind die selben wie bei der Anmeldung am delight Programm.

**Einzelbenutzer-Version:** Benutzername und Passwort müssen leer sein.

### 4.1 WDSL

Dieses Kapitel richtet sich an **Systemtechniker und Experten**

Wenn Sie Anwender sind, ist das sogenannte WDSL für Sie nicht von Interesse.

Das WDSL des legacy SOAP-Webservices kann über `{Verbindungs-URL}?wdsI` (zB. <http://localhost:8090/?wdsI>) aufgerufen werden.

**Wichtig:** Für neue Anwendungen verwenden Sie bitte die JSON REST-API. Die SOAP API ist deprecated seit v2021.

# Index

## - E -

Erlaubte API Methoden 16

## - M -

Methoden 16

## - O -

öffentlich 16

## - P -

publicdatauser 16

## - S -

Support 16

## - T -

Ticketsystem 16