



Konfigurationsanleitung – CSI | Database Exporter for ELO

Stand: 04/2022

Titel: Konfigurationsanleitung – CSI | Database Exporter for ELO
Kategorie: Anleitung
Aktuelle Version: Version 1.10

Allgemeines

Irrtümer, Änderungen und Druckfehler vorbehalten. Alle Angaben sind unverbindlich, die technischen Angaben entsprechen den Herstellerangaben. Keine Haftung und Gewähr bei unzutreffenden Informationen, fehlerhaften und unterbliebenen Eintragungen. Die Wiedergabe von Firmennamen, Produktnamen und Logos berechtigt nicht zu der Annahme, dass diese Namen/Bezeichnungen ohne Zustimmung der jeweiligen Firmen von jedermann genutzt werden dürfen. Es handelt sich um gesetzlich oder vertraglich geschützte Namen/Bezeichnungen, auch wenn sie im Einzelfall nicht als solche gekennzeichnet sind.

Es gelten ausschließlich die AGB der CSI GmbH. Diese finden Sie auch im Internet unter www.csi.gmbh.

Rechtlicher Hinweis

Das Dokument einschließlich aller damit verbundenen Unterlagen ist ausschließlich für Kunden und Partner von CSI GmbH bestimmt. Es darf ohne schriftliche Zustimmung seitens CSI GmbH nicht an Personen außerhalb der genannten Unternehmen weitergegeben und in keiner Form, außer um eine unternehmensinterne Einweisung in das behandelte Thema zu ermöglichen, kopiert oder reproduziert werden.

Dieses Dokument kann Teil eines Angebots sein. Ein rechtsverbindlicher Vertrag kommt erst durch die gemeinsame Unterzeichnung einer Vertragsurkunde oder durch die schriftliche Annahme eines Kundenauftrags durch CSI GmbH zustande.

Alle Rechte vorbehalten.

©CSI GmbH 2022

Inhalt:

1	Installation.....	4
1.1	Voraussetzungen.....	4
1.2	Einspielen der Lizenzen.....	4
1.3	Installation CSI Database Exporter for ELO.....	4
2	Konfiguration.....	5
2.1	Verwendung des WF-Node Bausteins im Workflow.....	6
2.2	Verwendung der RegisteredFunction.....	7
2.3	Aufbau der Relationstabellen.....	7
2.3.1	Konfiguration von Relationstabellen.....	8
3	Abbildungsverzeichnis.....	9



1 Installation

1.1 Voraussetzungen

-

1.2 Einspielen der Lizenzen

Die Lizenzen für die Solutions stellen wir Ihnen als eloinst-Dateien bereit. Das Einspielen der Lizenzen erfolgt dementsprechend über Drag-and-Drop in Ihr ELO Archiv.

Eine Übersicht der eingespielten Lizenzen finden Sie CSI LicenseManager, welcher als Kachel zur Verfügung steht (oder alternativ im Overview-Ordner der BusinessSolution "csi.common"). Im Standard ist diese Übersicht nur für ELO Administratoren sichtbar.

1.3 Installation CSI | Database Exporter for ELO

Die Installation der Business Solutions erfolgt über eine eloinst-Datei per „Drag and Drop“ in den ELO Java Client.

Die eloinst-Datei enthält dabei alle für das Produkt benötigten Solutions im Bundle.

Nach der Installation ist ein Neustart der ELO-Dienste erforderlich.

2 Konfiguration

Alle Konfigurationen sind grundsätzlich nur im Business Solutions **Custom**-Bereich (*Administration // Business Solutions Custom*) durchzuführen. Änderungen, die direkt im Produktbereich durchgeführt werden, gehen bei Updates verloren.

Sofern durch die Solution Workflows mit ausgeliefert werden, empfehlen wir, die Workflowtemplates zu kopieren und mit einem eigenen Namen zu speichern. Ansonsten würden auch eigene Anpassungen in den Workflows bei Updates überschrieben werden.

2.1 Verwendung des WF-Node Bausteins im Workflow

Der Baustein wird als Ende-Skript innerhalb eines Workflowknotens verwendet. Die Zuweisung zu dem Mappingprofil erfolgt dabei in den Skripteigenschaften des Knotens.

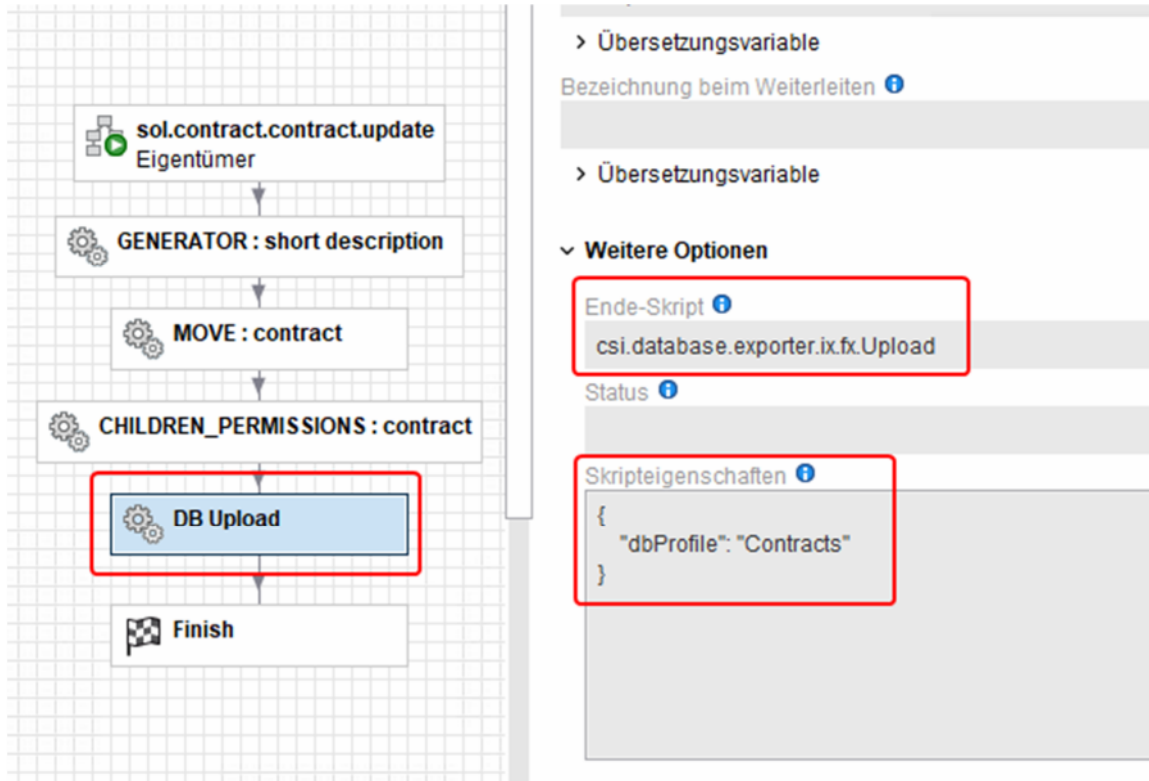


Abbildung 1 - Skriptaufruf im ELO Workflow

Ende-Skript:

csi . database . exporter . i x . fx . Upl oad

Eigenschaften:

```
{
  "dbProfile": ""
}
```

2.2 Verwendung der RegisteredFunction

Der Uploadvorgang kann ebenso in einem Skript (AS, IX, JC) untergebracht werden durch den Aufruf der Registered Function.

Dabei wird die objId sowie das dbProfile benötigt. Optional kann der Parameter ConfigPath angegeben werden, falls es sich um eine externe Konfigurationsdatei handelt.

```
sol.common.IxUtils.execute('RF_csi_dbExporter_Update', {  
  objId: 6857,  
  dbProfile: "Contracts",  
  configPath: "/csi.database.exporter/Configuration/contact.exporter.config"  
});
```

2.3 Aufbau der Relationstabellen

Eine Relationstabelle wird für die Darstellung von Multiline- sowie Multiindexfelder verwendet.

Dabei ist zu beachten, dass pro Multilinekonstrukt eine eigene Relationstabelle angelegt wird und diese nicht mit den Multiindextabellen gemischt werden.

Der Name der Tabelle ergibt sich aus dem Namen der Haupttabelle verbunden mit dem Namen der Relationstabelle durch ein "_". (z. B. CSI_Invoice_Positions). Dies kann in der Relationstabellenkonfiguration überschrieben werden.

Die Relationstabellen müssen immer eine Spalte gleichnamig der Primarykey-Spalte besitzen sowie eine Spalte "LineNo".

- Konvertieren: Der Wert wird in eine ganze Zahl konvertiert und auch als Zahl in die Tabelle eingetragen.
- dontQuoteName: Bei aktivierter Einstellung werden die Tabellennamen nicht wie im Standard in den SQL-Statements gequoted. Dies ermöglicht es, mit Datenbanken, die Case-Sensitive sind, und mit zusammengesetzten Namen wie app_global.mandator zu arbeiten.
- Echten Primärschlüssel verwenden: Bei gesetzter Option wird statt des als Primärschlüssel im Metadaten-Mapping definierten Wertes der Wert einer anderen Spalte (Einstellung Primärschlüsselspalte) als Primärschlüssel für das Eintragen in Relationstabellen verwendet. Dies ermöglicht es zum Beispiel, die Mandantennummer als Primärschlüssel zum Identifizieren für den Exporter zu verwenden, für die Relationstabelle "mandator_adress" aber den echten Primärschlüssel "id" zu verwenden.
- Primärschlüsselspalte: Bei aktivierter Einstellung "Echten Primärschlüssel verwenden" kann hier der Name der Primärschlüsselspalte angegeben werden. Wenn hier nichts eingetragen wird und die Einstellung aktiv ist, wird der Spaltenname "id" verwendet.

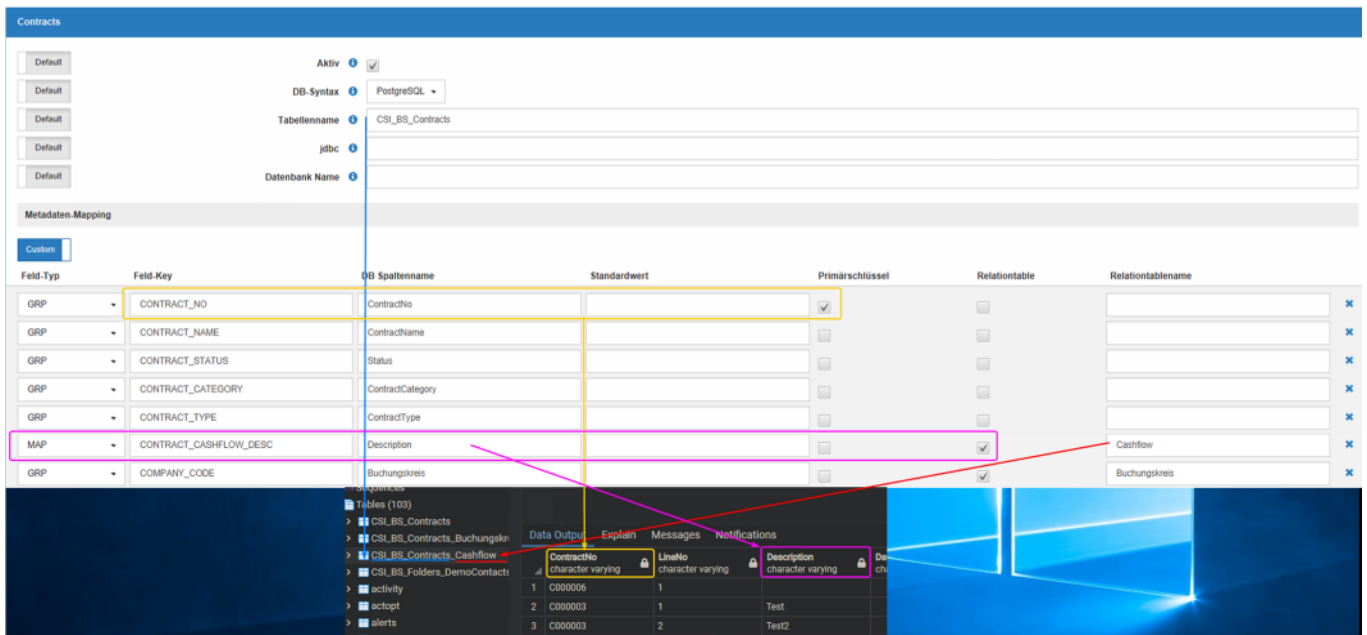


Abbildung 2 - Mappingbeispiel für Vertragsdaten

2.3.1 Konfiguration von Relationstabellen

Hier können zusätzliche Einstellungen zu den Relationstabellen getätigt werden.

- Relationstabelle: Der Name der Relationstabelle muss identisch mit dem Namen aus der Spalte "Relationstabelle" im Metadaten-Mapping sein.
- useOnlyName: Steuert, ob der Namen aus der Spalte "Relationstabelle" im Metadaten-Mapping so wie eingetragen verwendet werden soll (Haken ist gesetzt) oder ob der Tabellename wie oben beschrieben zusammengesetzt werden soll (kein Haken).
Bei dem Beispiel der Relationstabelle "CSI_BS_Contracts_Cashflow", zusammengesetzt aus "CSI_BS_Contracts" und "Cashflow", würde bei gesetztem Haken nicht "CSI_BS_Contracts_Cashflow" der Name der Tabelle sein sondern nur "Cashflow".
- primaryKey: Der Name der Datenbankspalte des Primärschlüssels in der Relationstabelle. Dadurch muss die Spalte nicht denselben Namen haben wie in der Ursprungstabelle (als Primärschlüssel in dem Metadaten-Mapping markiert) und kann stattdessen einen anderen Namen haben.

Wenn kein Eintrag für eine Relationstabelle existiert, werden folgende Standardwerte verwendet:

- useOnlyName: false. Es wird der oben beschriebene zusammengesetzte Name verwendet.
- primaryKey: Es wird der Name der als Primärschlüssel markierten Zeile im Metadaten-Mapping verwendet. Somit muss der Primärschlüssel der Relationstabelle identisch sein mit der Ursprungstabelle.

3 Abbildungsverzeichnis

Abbildung 1 - Skriptaufruf im ELO Workflow	6
Abbildung 3 - Mappingbeispiel für Vertragsdaten	8