

Dokumentation & Installation
Schnittstelle
Prozessanbindung

Installations- und Benutzerhandbuch
Schleupen SE

| | |
|------------------|----------------------------|
| Autor | Kambach, Vincent |
| Stand | 24.05.2023 |
| Version | V 1.0.4.0 |
| Status | freigegeben |
| Firma | Schleupen SE |
| Dokumentenpflege | Kambach, Vincent |
| Dateiname | Benutzerdokumentation.docx |

Impressum

Herausgegeben von: Schleupen SE
EDV-Partner für ausgewählte Branchen
Galmesweg 58
47445 Moers
Tel +49 (0) 2841 912-0
Fax +49 (0) 2841 998 283 0
www.schleupen.de

Zuständig für den Inhalt: Schleupen SE

Herstellung und Druck: Erstellt mit Microsoft ® Office Word 2016,
Juni 2023

© Copyright: Schleupen SE, Galmesweg 58, 47445 Moers

Möglicherweise weist das vorliegende Dokument noch Druckfehler oder drucktechnische Mängel auf. In der Dokumentation verwendete Software-, Hardware- und Herstellerbezeichnungen sind in den meisten Fällen auch eingetragene Warenzeichen und unterliegen als solche den gesetzlichen Bestimmungen.

Inhaltsverzeichnis

| | | |
|-----------|--|----|
| 1 | Installation..... | 4 |
| 1.1 | Systemvoraussetzungen | 4 |
| 1.2 | Installation | 4 |
| 1.3 | Vorbereitungen Schleupen.CS | 7 |
| 1.3.1 | Voraussetzungen | 7 |
| 1.3.2 | Lizenierung | 7 |
| 1.3.3 | Einrichtung der Datenbank | 7 |
| 1.3.4 | Einrichtung des Vertragsdatenmanagers | 7 |
| 1.3.5 | Konfiguration der Schnittstelle..... | 9 |
| 2 | Schnittstellenbeschreibung | 10 |
| 2.1 | Infrastruktursicht..... | 10 |
| 2.1.1 | Systematische Darstellung..... | 10 |
| 2.2 | Bereitgestellter Webservice | 11 |
| 2.2.1 | Beschreibung der Datenstruktur..... | 11 |
| 2.2.1.1 | Anlage von Personen..... | 11 |
| 2.2.1.2 | Anlage von Verträgen für Lieferantenwechsel und Neuanmeldung | 17 |
| 2.2.1.3 | Importieren von Zählerständen | 26 |
| 2.2.1.4 | Änderung von Personen..... | 30 |
| 2.2.1.5 | Änderung von Televerbindungen | 34 |
| 2.2.1.6 | Änderung von Bankverbindungen..... | 37 |
| 2.2.1.7 | Änderung von Anschriften | 41 |
| 2.2.1.8 | Änderung von Abschlägen | 44 |
| 2.2.1.9 | Änderung von Rechnungseinheiten | 48 |
| 2.2.1.10 | Verwalten von Eigentümern an Abnahmestellen | 53 |
| 2.2.1.11 | Abfrage des Vorgangsstatus | 55 |
| 2.2.1.12 | Abfrage des Geschäftsprozessstatus..... | 58 |
| 2.2.1.13 | Wechsel von Tarifen | 62 |
| 2.2.1.14 | Auflösung von Verträgen | 65 |
| 2.2.1.15 | Setzen von Zahlsperren | 68 |
| 2.3 | Detailliertes Fehler- und Diagnoseprotokoll..... | 71 |
| Anhang A. | Support für Individualprojekte..... | 72 |
| Anhang B. | Versionshistorie..... | 73 |

1 Installation

1.1 Systemvoraussetzungen

Die **Schnittstelle Prozessanbindung** setzt einen installierten und konfigurierten Schleupen.CS-Applikationsserver der Sommerversion 2022 oder aktueller voraus. Weitere Hinweise zu Systemvoraussetzungen und Installation entnehmen Sie bitte dem Installationshandbuch und den Systemvoraussetzungen für Schleupen.CS (verfügbar im Downloadbereich des Kundencenters).

1.2 Installation

Die Installation wird durch den Start des Installationspaketes „Schleupen SchnittstelleProzessanbindung_X.X.X.msi“ gestartet. Daraufhin werden Sie vom Installationsassistenten begrüßt. Dieser führt Sie durch die einzelnen Schritte des Installationsvorganges.

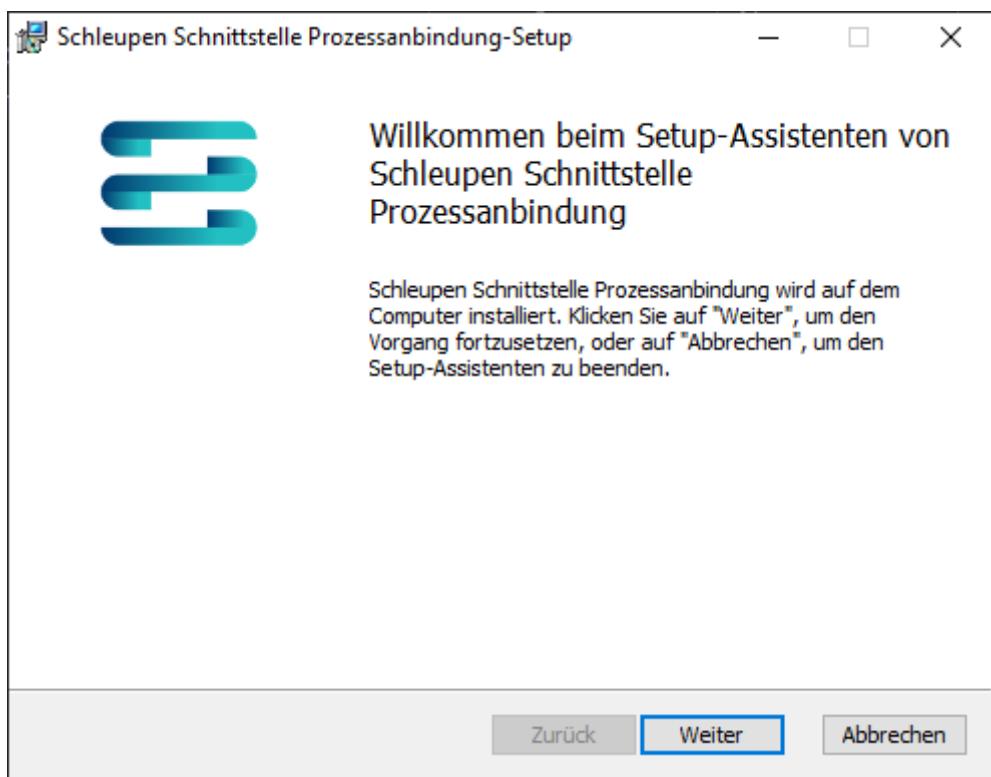


Abbildung 1 - Installationsassistent

Da die Software ein installiertes Schleupen.CS-System voraussetzt, werden die Komponenten der **Schnittstelle Prozessanbindung** in diesen Pfad eingebunden. Bei einer Standardinstallation erfolgt dies unter:

`C:\Programme\Schleupen\SchnittstelleProzessanbindung`.

Im nächsten Installationsschritt werden die Produktfunktionen ausgewählt, welche installiert werden sollen.

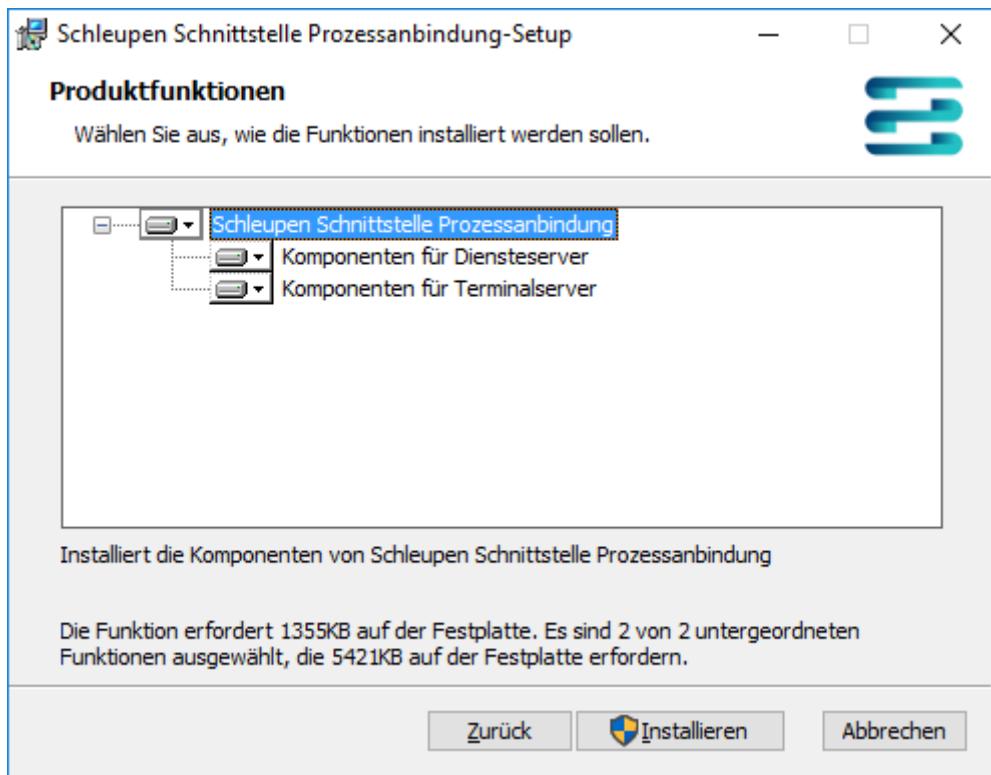


Abbildung 2 - Auswahl der Komponenten

Als Komponenten stehen zur Auswahl:

- **Terminalserver-Komponenten**

Hierbei handelt es sich um die Frontendkomponenten, welche der Konfiguration dienen. Diese Produktfunktion muss auf Terminal-Servern und/oder Clients installiert werden.

- **Dienstserver-Komponenten**

Hierbei handelt es sich um die Programmkomponenten, welche die Verarbeitung der durch den Job abgerufenen Daten und die Statusrückmeldung beinhalten. Diese Produktfunktion muss zwingend auf alle Dienste-Server installiert werden, welche im Schleupen.CS für Anbindung externer Produkte eingerichtet sind.

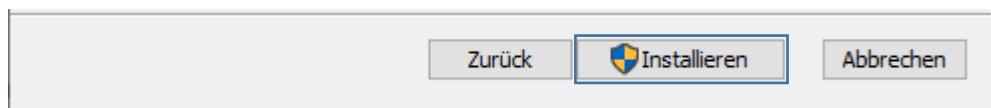


Abbildung 3 - Installation bestätigen

Wird die Installation bestätigt, werden die ausgewählten Komponenten auf dem System installiert, notwendige Einstellungen durchgeführt und ggf. Verknüpfungen für den Start der Anwendung angelegt.

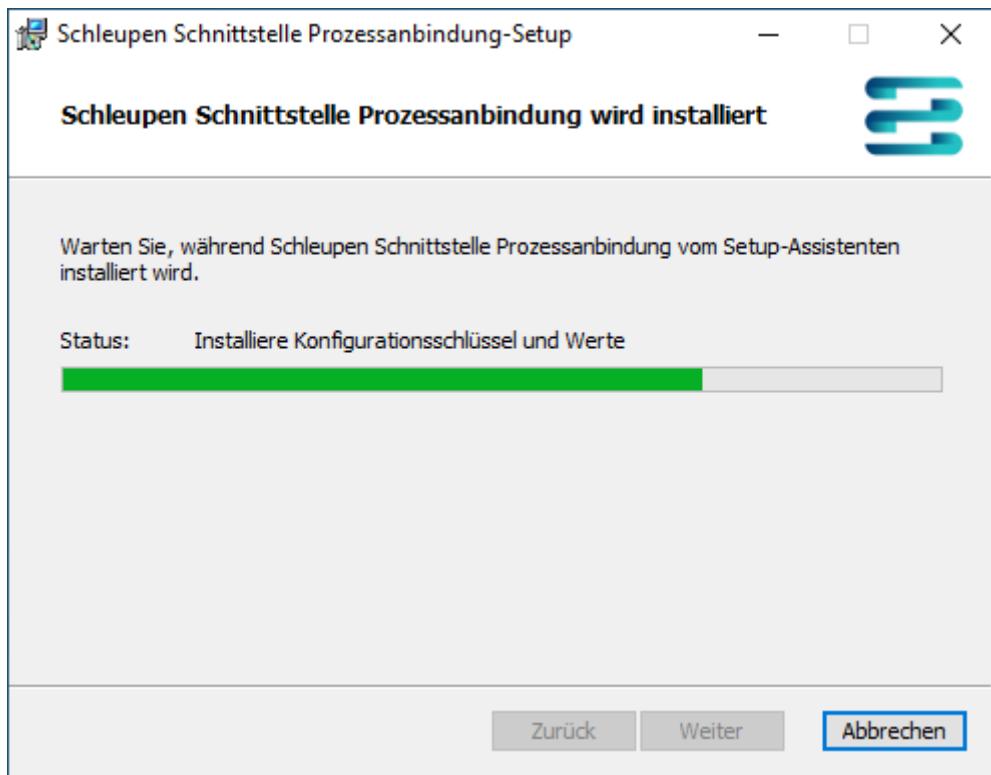


Abbildung 4 - Installationsfortschritt

Der Assistent wird zum Abschluss über die Schaltfläche **Fertig stellen** beendet.

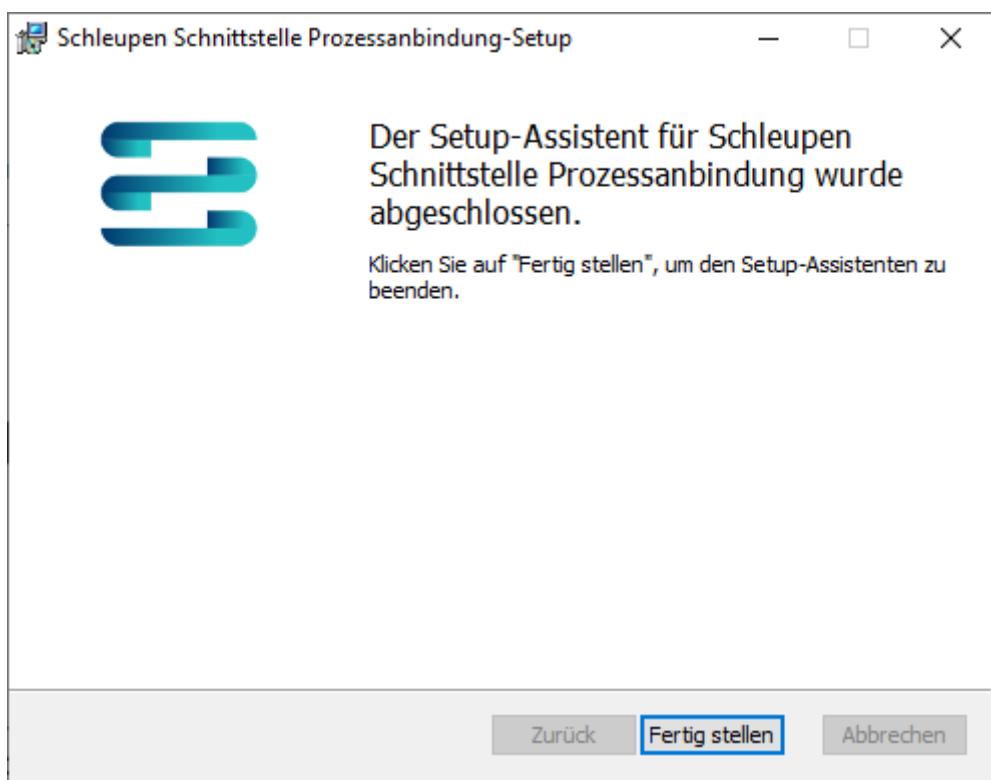


Abbildung 5 - Installation fertiggestellt

1.3 Vorbereitungen Schleupen.CS

1.3.1 Voraussetzungen

Um die Schnittstelle einsetzen zu können sind folgende Voraussetzungen notwendig:

- Ablage einer passenden Lizenzdatei (siehe 1.3.2)
- Einrichtung der Datenbank (siehe 1.3.3)
- Einrichtung des Vertragsdatenmanagers (siehe 1.3.4)
- Konfiguration der Schnittstelle (siehe 1.3.5)

1.3.2 Lizensierung

Für die Schnittstelle Prozessanbindung ist eine separate Lizenz erforderlich. Diese wird in der Lizenzdatei für Individualprodukte bereitgestellt. Die Lizenzdatei muss in das Verzeichnis *Licenses* im Installationspfad der Anwendung hinterlegt werden.

Alternativ kann die Lizenzdatei in das Unterverzeichnis *Individuallizenzen* im Verzeichnis *\Dat\Licenses* der Schleupen.CS-Installation abgelegt werden.

Der vollständige Pfad lautet bei der Standardinstallation:

C:\Programme\Schleupen\Dat\Licenses\Individuallizenzen.

Dieses Vorgehen bietet sich vor allem dann an, wenn entweder mehrere Individualprodukte im Einsatz sind oder das Individualprodukt auf mehreren Applikationsservern installiert wird. Erfolgt die Konfiguration der Applikationsserver über ein globales Ablageverzeichnis für die Konfiguration, so muss dieses genutzt werden.

1.3.3 Einrichtung der Datenbank

Die Verarbeitung der Schnittstelle erfolgt über einen Schleupen.CS System, welches über das Basissystem eingerichtet und gesteuert wird. Weiterhin werden mehrere Datenbanktabellen in der Schleupen.CS Datenbank zur Ablage von Verarbeitungsinformationen verwendet.

Hierfür ist eine Ersteinrichtung aller Datenbankobjekte einmalig pro Datenbank notwendig.

Die Einrichtung der Datenbank erfolgt über die Ausführung des Installationsskriptes *Datenbank_einrichten.mss.sql* im Ordner *SQL* des Installationsverzeichnisses.

Die vom Installationsskript durchgeführten Änderungen können durch das Skript *Datenbankeinrichtung_entfernen.mss.sql* wieder vollständig rückgängig gemacht werden.

1.3.4 Einrichtung des Vertragsdatenmanagers

Für Vertragsanlage- und Auflösungsvorgänge, die Änderung von Personenstammdaten und weitere Prozesse ist die Einrichtung einer Datenherkunft erforderlich. Die Einstellung erreichen Sie in CS.NM Netzzugangsmanagement über → Menü Einstellungen → Vertragsdatenmanager.



Abbildung 6 - Einstellungen CS.NM

In den Einstellungen können verschiedene Datenherkünfte unterschiedlicher Versionen bereitgestellt werden. Diese dienen später bei der Verarbeitung der Prozesse durch den VDM für die Unterscheidung, vor allem bei der Vertragsdatenergänzung. Einzelne Funktionen der Schnittstelle bieten die Angabe der Datenherkunft beim Aufruf an. Sollte hier keine Angabe erfolgen, so wird als Standardwert „PORTAL“ mit der Version „1.0“ verwendet. Die Vertragsdatenergänzung ist für die Funktionsweise des VDM vor allem für die Anlage von Verträgen wichtig.

- ! Für die Durchführung von Vertragsanlagevorgängen ist die Einrichtung von Vertragsdatenergänzungen erforderlich.
- Weitere Details zur Konfiguration der Datenherkunft sowie Ausführungen zur Konfiguration der Vertragsdatenergänzung entnehmen Sie bitte der Hilfe bzw. dem Installationshandbuch von Schleupen.CS.

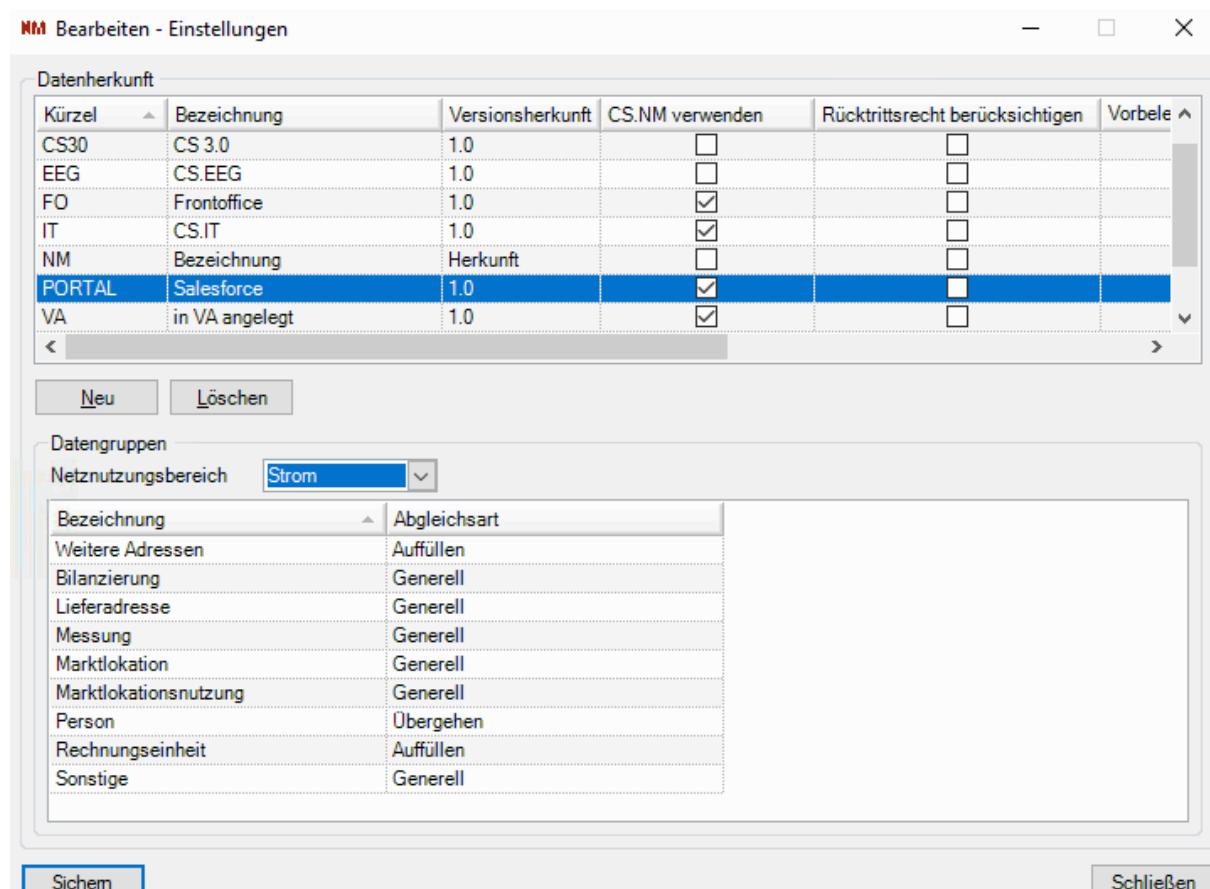


Abbildung 7 - Datenherkunft

Die Abgleichseinstellungen der Datengruppen müssen so vorgenommen werden, dass die im Zuge der Marktkommunikation ausgetauschten Informationen korrekt an den Vorgang übernommen werden. Sollte das Produkt CS.FO oder CS.IT im Einsatz sein, so kann die Datenherkunft dieser Produkte als Vorlage dienen, da die Vorgänge vergleichbar verarbeitet werden.

1.3.5 Konfiguration der Schnittstelle

Die Konfiguration erfolgt im Basissystem unter der Kategorie [*Konfigurationsverwaltung>Konfigurationen*](#). Hier werden folgende Parameter gesetzt (Pflichtfelder sind **Fett** gedruckt):

- **Lieferanten** (Zuweisung welches Kürzel zu welchem Energiehändler gehört)

2 Schnittstellenbeschreibung

2.1 Infrastruktursicht

2.1.1 Systematische Darstellung

Das Gesamtsystem besteht aus mehreren Hauptkomponenten, welche miteinander über ein Netzwerk kommunizieren. Auf der Seite von Schleupen.CS werden Services angeboten, welche durch einen Client konsumiert werden.

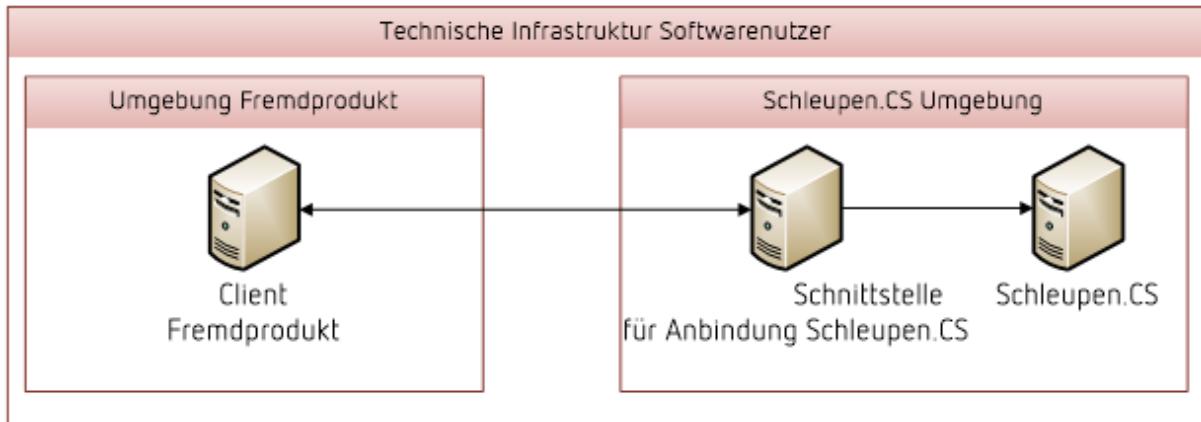


Abbildung 8 – Beispiel Infrastruktur – vertrauenswürdige Intranetumgebung



Abbildung 9 - Beispiel Infrastruktur – Übertragung über öffentliche Netzwerke

2.2 Bereitgestellter Webservice

2.2.1 Beschreibung der Datenstruktur

Nachfolgend werden die Webservices und die dazugehörigen Datenmodelle beschrieben. Die Angabe der Aufrufadressen erfolgen zur Vereinfachung immer nur mit dem Schema „http“. Der Aufruf ist bei entsprechend eingerichteter Infrastruktur gleichermaßen verschlüsselt über das Schema „https“ möglich.

2.2.1.1 Anlage von Personen

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Person>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Person>

Diese Anfrage legt in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ eine neue Person an. Ein Beispiel für den Datencontainer wird im Konzept später dargestellt.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|-------------|--|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| Person | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja | Die Stammdaten der Person, welche angelegt werden soll |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Person

Content-Type: application/sag.personanlage.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|-------------------|---------|---|
| Art | Zeichenkette (1) | Ja | <p>Angabe, ob es sich um eine natürliche oder juristische Person handelt. Mögliche Werte:</p> <ul style="list-style-type: none"> • „J“ – juristische Person • „N“ – natürliche Person |
| Name | Zeichenkette (70) | Ja | Name der Person |
| Name2 | Zeichenkette (70) | Nein | Zweite Namensangabe bei juristischen Personen |
| Vorname | Zeichenkette (70) | Nein | Vorname der Person, nur bei natürlichen Personen Pflicht |
| Zusatz1 | Zeichenkette (70) | Nein | Zusatz 1 zum Namen |
| Zusatz2 | Zeichenkette (70) | Nein | Zusatz 2 zum Namen |
| Anrede | Zeichenkette (4) | Nein | <p>Angabe des Kürzels einer im Schleupen.CS System verwalteten Anrede. Folgende Werte sind bspw. möglich:</p> <ul style="list-style-type: none"> • Fr • Hr • Fam <p>Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten.</p> |
| Titel | Zeichenkette (8) | Nein | <p>Angabe des Kürzels eines im Schleupen.CS System verwalteten Titels. Folgende Werte sind bspw. möglich:</p> <ul style="list-style-type: none"> • Dr • ProfDr • RA |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|------------------|---|---------|--|
| | | | Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten. |
| Geburtsdatum | Datum | Nein | <p>Angabe des Geburtsdatums bei natürlichen Personen</p> <p>Format: yyyy-MM-dd</p> <p>Beispiel: 2019-12-31</p> |
| Anschriften | Liste von Objekten vom Typ Anschrift | Ja | Eine Liste der Anschriften der Person, es muss mindestens eine Anschrift übergeben werden |
| Televerbindungen | Liste von Objekten vom Typ Televerbindung | Nein | Optionale Liste von Televerbindungsdaten (Telefon, Email, ...) |

Anschrift

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------------|-------------------|----------|--|
| PLZ | Zeichenkette (10) | Ja | Postleitzahl der Anschrift |
| Ort | Zeichenkette (40) | Ja | Ortsangabe der Anschrift |
| Ortsteil | Zeichenkette (40) | Nein | Angabe des Ortsteiles |
| Strasse | Zeichenkette (40) | Nein (1) | Straßenangabe der Anschrift |
| Hausnummer | Zeichenkette (8) | Nein (1) | Hausnummernangabe der Anschrift |
| Hausnummernzusatz | Zeichenkette (4) | Nein | Der Zusatz zur Hausnummer |
| Hausnummernteil | Zeichenkette (8) | Nein | Zusätzliche Angabe zur Hausnummer |
| Postfach | Zeichenkette (10) | Nein (1) | Angabe des Postfachs |
| Land | Zeichenkette (4) | Nein | Landeskürzel der Anschrift (z.B. D für Deutschland) Standardwert: D |
| Art | Zeichenkette (4) | Nein | Angabe der Art der Anschrift als Kürzel laut Verwaltung der Stammdaten in Schleupen.CS. Mögliche Werte sind bspw. „AA“ – Allgemeine Anschrift „RE“ – Rechnungsanschrift „LI“ – Lieferanschrift Die vollständige Liste ist abhängig vom Customizing der jeweiligen Datenbank |

Hinweise:

(1) Anschriften können in drei verschiedenen Ausprägungen vorhanden sein:

- Normale Anschrift: Angabe von Straße und Hausnummer, Postfach ist nicht gefüllt
- Postfachanschrift: Angabe des Postfachs, Straße, Hausnummer und Hausnummernzusatz ist nicht gefüllt
- Großkundenanschrift: es ist weder Straße, Hausnummer und Hausnummernzusatz noch ein Postfach angegeben

Televerbindung

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|--------------------|---------|--|
| Art | Zeichenkette (8) | Ja | <p>Angabe der Art der Televerbindung als Kürzel laut Verwaltung der Stammdaten in Schleupen.CS. Mögliche Werte sind bspw.</p> <ul style="list-style-type: none"> • „Tel“ – Telefon • „e-mail“ – Email-Adresse • „mobil“ – Mobilfunknummer <p>Die vollständige Liste ist abhängig vom Customizing der jeweiligen Datenbank</p> |
| Nummer | Zeichenkette (120) | Ja | Nummer bzw. Adresse der Televerbindung, z.B. „987654321“, „0351/643222“ oder „max.mustermann@web.de“ |
| Vorwahl | Zeichenkette (50) | Nein | Optionale Angabe der Vorwahl zur Nummer, wenn diese getrennt vorliegt bzw. getrennt übernommen werden soll |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde die Person angelegt. Im http Header „Location“ der Antwortnachricht ist die relative Adresse zum Abruf der Person enthalten. Die relative Adresse ist nach folgendem Muster aufgebaut: </Schleupen/IVP/SchnittstelleProzessanbindung/Person/{PIN}>

Da diese Schnittstelle den Abruf der Stammdaten nicht vorsieht, antwortet der GET-Service nur mit dem Status 204 (NO CONTENT).

- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für die minimalistische Personenanlage:

Nachfolgend ist ein Beispiel für die Anlage einer Person als Firma mit Großkundenanschrift. Dieser Datencontainer stellt den minimalsten möglichen Umfang dar.

Content-Type: application/sag.personenanlage.v1+json

```
{  
    "Art": "N",  
    "Name": "Mustermann",  
    "Anschriften": [  
        {  
            "PLZ": "01067",  
            "Ort": "Musterstadt",  
            "Strasse": "Musterstrasse",  
            "Hausnummer": 4  
        }  
    ]  
}
```

Beispieldatensatz für die Personenanlage einer natürlichen Person mit Anschrift und Telefon:

Nachfolgend ist ein Beispiel für die Anlage einer natürlichen Person mit häufig vorkommendem Umfang an Daten dargestellt.

Content-Type: application/sag.personenanlage.v1+json

```
{  
    "Art": "J",  
    "Name": "Mustermann",  
    "Vorname": "Martha",  
    "Anrede": "Fr",  
    "Anschriften": [  
        {  
            "PLZ": "01067",  
            "Ort": "Musterstadt",  
            "Strasse": "Musterstrasse",  
            "Hausnummer": "1",  
            "Art": "AA"  
        }  
    ],  
    "Televerbindungen": [  
        {  
            "Art": "Tel",  
            "Vorwahl": "0815",  
            "Nummer": "988743"  
        },  
        {  
            "Art": "e-mail",  
            "Nummer": "martha.mustermann@web.de"  
        }  
    ]  
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

Content-Type: application/sag.fehlermeldung.v1+json

```
{  
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder  
    Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem  
    aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",  
    "Status": "404"  
}
```

2.2.1.2 Anlage von Verträgen für Lieferantenwechsel und Neuanmeldung

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Vertragsanlage>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Vertragsanlage>

Diese Anfrage erzeugt einen neuen Prozess zur Vertragsanlage. Im Body der Anfrage werden entsprechende Daten übergeben werden.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|----------------|--|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| Vertragsanlage | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja | JSON-formatierte Datenstruktur einer Vertragsanlage, für welche ein Prozess angelegt werden soll |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Vertragsanlage

Content-Type: application/sag.vertragsanlage.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|---------------------|-----------------|---------|--|
| Abnahmestelle | Abnahmestelle | Ja | Adressdaten der Abnahmestelle |
| Vertragspartner | Person | Ja | Personendaten des Vertragspartners |
| AllgemeineAnschrift | AnschriftObjekt | Nein | Optionale Angabe einer allgemeinen Anschrift des Vertragspartners, wenn diese von der Rechnungsanschrift abweicht. |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|---------------------------------|--------------------------|---------|--|
| | | | Weiterhin kann hier die Anschrift des Vertragspartners angegeben werden, wenn ein abweichender Rechnungsempfänger angegeben ist und somit die Rechnungsanschrift nicht dem Vertragspartner zugeordnet wird. Wenn ein abweichender Rechnungsempfänger übergeben wird, jedoch keine allgemeine Anschrift des Vertragspartners, so wird die Anschrift der Lieferstelle als allgemeine Anschrift des Vertragspartners genutzt. |
| AbweichenderRechnungsempfaenger | Person | Nein | Personendaten eines abweichenden Rechnungsempfängers |
| Rechnungsanschrift | Anschrift | Ja | Anschriftdaten des Vertragspartners; wenn AbweichenderRechnungsempfänger gefüllt ist, dann dessen Anschriftdaten |
| RechnungseinheitId | Ganzzahl | Nein | Optionale Angabe der Id einer Rechnungseinheit, wenn eine bestehende Rechnungseinheit für die Vertragsanlage genutzt werden soll |
| Bankverbindungen | Liste von Bankverbindung | Nein | Angabe der Bankverbindungsdaten für Abbucher |
| Vertragsbeginn | Datum | Ja/Nein | Datum, ab wann die Belieferung erfolgen soll; Wenn kein Wert angegeben wird, so erfolgt die Ermittlung anhand des nächstmöglichen Termins auf Basis der Kündigungsanfrage beim Vorlieferanten. Bei einer Neuanmeldung bzw. eines Lieferantenwechsels mit bereits erfolgter Kündigung muss der Vertragsbeginn angegeben werden. |
| Geschaeftsbereich | Zeichenkette (4) | Ja | Die Angabe des Geschäftsbereichskürzels laut Schleupen.CS |
| Tarifkuerzel | Zeichenkette (8) | Ja | Kürzel des Tarifs, welcher für den Vertragsabschluss genutzt werden soll |
| Vorlieferant | Zeichenkette (35) | Ja/Nein | Angabe der Marktnummer (VDEW, ILN) des Vorlieferanten. Bei Neueinzug erfolgt hier keine Angabe. Für einen Lieferantenwechsel ist es ein Pflichtfeld. |
| VorlieferantEIC | Zeichenkette (16) | Nein | Angabe des Bilanzkreises des Vorlieferanten. Wenn hier keine Angabe erfolgt, müssen die Energiehändlerkonten in Schleupen.CS eindeutig über die Identifikationsnummer ermittelbar sein. |
| VorlieferantKundennummer | Zeichenkette (70) | Nein | Kundennummer beim Vorlieferant |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-----------------------|-------------------|---------|--|
| VorlieferantKuendigen | Zeichenkette (1) | Nein | <p>Gibt an, ob dem Vorlieferant eine Kündigung geschickt werden soll.</p> <ul style="list-style-type: none"> • Wert „J“: Kündigung auslösen, Bedingung, es handelt sich um einen Lieferantenwechsel, d.h. das Element Vorlieferant ist gefüllt • Wert „N“: keine Kündigung auslösen, ebenfalls nur bei einem Lieferantenwechsel relevant; in diesem Fall ist bspw. die Kündigung durch den Kunden selbst bereits erfolgt <p>Standardwert: „J“, wenn Vorlieferant gefüllt ist</p> |
| Zaehlernummer | Zeichenkette (20) | Ja | Nummer des Gerätes, für welches der Vertrag abgeschlossen werden soll |
| PrognoseHT | Festkommazahl | Nein | Angabe des jüngsten prognostizierten Jahresverbrauchs ET bzw. HT |
| PrognoseNT | Festkommazahl | Nein | Angabe des jüngsten prognostizierten Jahresverbrauchs NT |
| Lieferantenschluessel | Zeichenkette (16) | Ja | Angabe eines Schlüsselwertes, welcher zur Identifikation des Energiehändlerkontos verwendet wird. Die Zuordnung eines Kontos zu einem Schlüsselwert erfolgt über einen Konfigurationsdialog (siehe Punkt 1.3.5). |
| Netzbetreiber | Zeichenkette (35) | Ja | Angabe der Marktnummer (VDEW, ILN) des Netzbetreibers |
| Transaktionsgrund | Zeichenkette (3) | Nein | <p>Angabe eines gültigen Kürzels laut Definition UTILMD für den Transaktionsgrund, welcher in der Marktkommunikation genutzt wird</p> <p>Wenn kein Wert übergeben wird, so wird</p> <ul style="list-style-type: none"> • E01 genutzt, wenn kein Wert im Feld Vorlieferant gesetzt ist • E03 genutzt, wenn ein Wert im Feld Vorlieferant gesetzt ist |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|----------------------|--|---------|---|
| Abrechnungsfeldwerte | Liste von Objekten vom Typ Abrechnungsfeldwert | Nein | Optionale Angabe von Abrechnungsfeldwerten, welche an den Vertrag übernommen werden sollen |
| Herkunft | Zeichenkette (35) | Nein | Optionale Angabe der Herkunft zur Identifikation der Vorgänge, welche aus dem Fremdsystem übernommen wurden. Der Standardwert ist „PORTAL“. |
| HerkunftVersion | Zeichenkette (10) | Nein | Optionale Angabe der Version zur Identifikation der Vorgänge, welche aus dem Fremdsystem übernommen wurden. Der Standardwert ist „1.0“. |

Abnahmestelle

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|-------------------|---------|---|
| Id | Ganzzahl | Ja/Nein | Technischer Schlüssel der Abnahmestelle, wenn bekannt Wenn ein Wert angegeben wird, dann ist keine Angabe der Fachdaten notwendig. Es wird die Abnahmestelle laut Id-Wert genutzt. Wenn kein Wert angegeben wird, so übernimmt der Vertragsdatenmanager die Identifikation der Abnahmestelle anhand der Fachdaten des Vorgangs und legt ggf. eine neue Abnahmestelle an |
| PLZ | Zeichenkette (10) | Ja/Nein | Postleitzahl der Abnahmestelle |
| Ort | Zeichenkette (40) | Ja/Nein | Ort der Abnahmestelle |
| Ortsteil | Zeichenkette (40) | Nein | Ortsteil der Abnahmestelle |
| Strasse | Zeichenkette (40) | Ja/Nein | Strasse der Abnahmestelle |
| Hausnummer | Zeichenkette (16) | Ja/Nein | Hausnummer der Abnahmestelle |

Person

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|------------|-------------|---|
| PIN | Ganzzahl | Ja/Nein (1) | Die PIN der Person |
| Art | Auflistung | Ja/Nein (1) | Angabe, ob es sich um eine natürliche oder juristische Person handelt. Mögliche Werte: <ul style="list-style-type: none">• „J“ – juristische Person• „N“ – natürliche Person |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|------------------|---|-------------|--|
| Name | Zeichenkette (70) | Ja/Nein (1) | Name der Person |
| Name2 | Zeichenkette (70) | Nein | Zweite Namensangabe bei juristischen Personen |
| Vorname | Zeichenkette (70) | Nein | Vorname der Person, nur bei natürlichen Personen Pflicht |
| Zusatz1 | Zeichenkette (70) | Nein | Zusatz 1 zum Namen |
| Zusatz2 | Zeichenkette (70) | Nein | Zusatz 2 zum Namen |
| Anrede | Zeichenkette (4) | Nein | <p>Angabe des Kürzels einer im Schleupen.CS System verwalteten Anrede. Folgende Werte sind bspw. möglich:</p> <ul style="list-style-type: none"> • Fr • Hr • Fam <p>Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten.</p> |
| Titel | Zeichenkette (8) | Nein | <p>Angabe des Kürzels eines im Schleupen.CS System verwalteten Titels. Folgende Werte sind bspw. möglich:</p> <ul style="list-style-type: none"> • Dr • ProfDr • RA <p>Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten.</p> |
| Geburtsdatum | Datum | Nein | <p>Angabe des Geburtsdatums bei natürlichen Personen</p> <p>Format: yyyy-MM-dd</p> <p>Beispiel: 2019-12-31</p> |
| Televerbindungen | Liste von Objekten vom Typ Televerbindung | Nein | Optionale Liste von Televerbindungsdaten (Telefon, Email, ...) |

(1) Bei der Übergabe der Person kann entweder eine bereits im System vorhandene Person referenziert werden oder es werden die vollständigen Fachdaten für die neu anzulegende Person übergeben. Im ersten Fall wird nur die PIN übermittelt, im zweiten Fall wird keine PIN angegeben und stattdessen mindestens Art und Name

Televerbindung

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|--------------------|---------|--|
| Art | Zeichenkette (8) | Ja | <p>Angabe der Art der Televerbindung als Kürzel laut Verwaltung der Stammdaten in Schleupen.CS. Mögliche Werte sind bspw.</p> <ul style="list-style-type: none"> • „Tel“ – Telefon • „e-mail“ – Email-Adresse • „mobil“ – Mobilfunknummer <p>Die vollständige Liste ist abhängig vom Customizing der jeweiligen Datenbank</p> |
| Nummer | Zeichenkette (120) | Ja | Nummer bzw. Adresse der Televerbindung, z.b. „987654321“, „0351/643222“ oder „max.mustermann@web.de“ |
| Vorwahl | Zeichenkette (50) | Nein | Optionale Angabe der Vorwahl zur Nummer, wenn diese getrennt vorliegt bzw. getrennt übernommen werden soll |

Anschrift

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|-------------------|---------|---|
| PLZ | Zeichenkette (10) | Ja | Postleitzahl der Anschrift |
| Ort | Zeichenkette (40) | Ja | Ortsangabe der Anschrift |
| Ortsteil | Zeichenkette (40) | Nein | Angabe des Ortsteiles |
| Strasse | Zeichenkette (40) | Nein | Straßenangabe der Anschrift |
| Hausnummer | Zeichenkette (20) | Nein | Hausnummernangabe der Anschrift |
| Postfach | Zeichenkette (10) | Nein | Angabe des Postfachs |
| Land | Zeichenkette (4) | Nein | <p>Landeskürzel der Anschrift (z.B. D für Deutschland)</p> <p>Standardwert: D</p> |

Hinweise:

(1) Anschriften können in drei verschiedenen Ausprägungen vorhanden sein:

- Normale Anschrift: Angabe von Straße und Hausnummer, Postfach ist nicht gefüllt
- Postfachanschrift: Angabe des Postfachs, Straße, Hausnummer und Hausnummernzusatz ist nicht gefüllt
- Großkundenanschrift: es ist weder Straße, Hausnummer und Hausnummernzusatz noch ein Postfach angegeben

Bankverbindung

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-----------------|-------------------|---------|---|
| IBAN | Zeichenkette (34) | Ja | Angabe der IBAN der Bankverbindung |
| Bankname | Zeichenkette (40) | Nein | Name der Bank |
| BLZ | Zeichenkette (10) | Nein | Angabe der Bankleitzahl Wenn es sich nicht um eine deutsche IBAN handelt, muss die Bankleitzahl angegeben werden. |
| Kontonummer | Zeichenkette (20) | Nein | Angabe der Kontonummer Wenn es sich nicht um eine deutsche IBAN handelt, muss die Kontonummer angegeben werden. |
| BIC | Zeichenkette (11) | Nein | Angabe der BIC der Bank Wenn es sich nicht um eine deutsche IBAN handelt, muss die BIC angegeben werden |
| Inhaber | Zeichenkette (40) | Nein | Angabe des Inhabers der Bankverbindung |
| GueltigAb | Datum | Nein | Datum, ab wann die Bankverbindung gültig ist und somit verwendet werden kann Format: yyyy-MM-dd Beispiele: 2019-12-31 Standardwert: Datum zum Zeitpunkt der Anlage des Prozesses |
| GueltigBis | Datum | Nein | Datum, bis wann die Bankverbindung gültig ist und somit verwendet werden kann Format: yyyy-MM-dd Beispiele: 2019-12-31 |
| Art | Zeichenkette (2) | Nein | Mögliche Werte: FG – Forderungen und Guthaben FO – nur Forderungen GH – nur Guthaben Default: FG |
| Zweck | Zeichenkette (2) | Nein | Mögliche Werte: AR – Abschläge und Rechnungen AB – nur Abschläge RE – nur Rechnungen Default: AR |
| AbwKontoinhaber | Person | Nein | Angabe des Kontoinhabers, wenn dieser vom Vertragspartner abweicht |

Abrechnungsfeldwert

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|--------------|-------------------|---------|--|
| Feldreferenz | Zeichenkette (10) | Ja | Feldreferenz, welche das Abrechnungsfeld des Tarifs identifiziert, für welches ein Wert an den Vertrag gesetzt werden soll |
| Wert | Zeichenkette (40) | Ja | Wert, welcher für dieses Abrechnungsfeld am Vertrag gesetzt werden soll |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde der Prozess für die Vertragsanlage angelegt. Im Antwort-Kopf wird im Element „Location“ die relative Adresse zur Abfrage des Prozessstatus übermittelt (siehe Punkt 2.2.1.12) /Schleupen/IVP/SchnittstelleProzessanbindung/Geschaeftsprozesstatus/{Id_des_Geschaeftsprozesses}
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für einen Vertragsanlageprozess:

Nachfolgend ist ein Beispiel für die Übermittlung der Daten im beschriebenen Format dargestellt, um einen neuen Prozess für die Vertragsanlage auszulösen:

```
Content-Type: application/sag.vertragsanlage.v1+json
{
    "Abnahmestelle": {
        "PLZ": "01067",
        "Ort": "Dresden",
        "Strasse": "Kramergasse",
        "Hausnummer": "2"
    },
    "Vertragspartner": {
        "Art": "J",
        "Name": "Schleupen AG"
    },
    "Rechnungsanschrift": {
        "PLZ": "01067",
        "Ort": "Dresden",
        "Strasse": "Kramergasse",
        "Hausnummer": "2"
    },
    "Vertragsbeginn": "2022-01-01",
    "Geschaeftsbereich": "S",
    "Tarifkuerzel": "FIX24",
    "Vorlieferant": "4041408000007",
    "VorlieferantKuendigen": "J",
    "Zaehlernummer": "EMH12345678901",
    "Lieferantenschluessel": "3536",
    "Netzbetreiber": "9900680000001",
    "Abrechnungsfeldwerte": [
        {
            "Feldreferenz": "VP.Schlüssel",
            "Wert": "01"
        }
    ]
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
    "Status": "404"
}
```

2.2.1.3 Importieren von Zählerständen

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Zaehlerstandsimport>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Zaehlerstandsimport>

Diese Anfrage erzeugt einen oder mehrere neue Importdatensätze zur Übernahme von Zählerständen an Abrechnungszählwerke von Geräten in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“.

Request

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|----------------|---|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der zu verwendenden Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (8) (Header) | Ja | Angabe des Kürzels des zu verwendenden Mandanten aus Schleupen.CS |
| Zaehlerstaende | Liste von Objekten vom Typ Zaehlerstand (im Nutzdatenbereich der Anfrage) | Ja | Fachliche Daten als JSON-formatiertes Objekt im Nutzdatenbereich der Anfrage |

Zaehlerstand

Content-Type: application/sag.zaehlerstand.v1+json

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------------|------------------|---------|---|
| Zaehlernummer | Zeichenkette (4) | Ja | Nummer des Zählers |
| Geschaeftsbereich | Zeichenkette (4) | Nein | Optionale Angabe des Geschäftsbereiches für die Zählersuche; diese Angabe muss erfolgen, wenn anhand der Zählernummer keine eindeutige Identifikation des Zählers möglich ist |

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------------------|---|---------|--|
| Zeitpunkt | Datum mit Zeitangabe | Ja | Ablesezeitpunkt der Zählwerksstände Format: yyyy-MM-ddThh:mmTZD Beispiele: 2019-12-31T11:30+02:00 |
| Zählwerksstände | Liste von Objekten vom Typ Zählwerksstand | Ja | Liste von Zählerwerken mit den jeweilig zu importierenden Ständen |
| Ablesegrund | Zeichenkette (2) | Nein | Optionale Angabe des Kürzels des Ablesegrundes aus den VA-Stammdaten. Default: „05“ (Zwischenablesung) Mögliche Werte: alle Ablesegründe, welche in den VA Stammdaten verwaltet sind. |
| Ablesekennzeichen | Zeichenkette (3) | Nein | Optionale Angabe des Kürzels des Ablesekennzeichens aus den VA-Stammdaten. Mögliche Werte: alle Ablesekennzeichen, welche in den VA Stammdaten verwaltet sind. Default: „001“ (Ablesung Kunde) |
| Ableseverantwortlicher | Zeichenkette (3) | Nein | Optionale Angabe des Kürzels des Ableseverantwortlichen aus den VA-Stammdaten. Mögliche Werte: <ul style="list-style-type: none">• „LIE“ – Lieferant• „MSB“ – Messstellenbetreiber• „VNB“ - Verteilnetzbetreiber Default: „VNB“ (Verteilnetzbetreiber) |
| BestehendeAktualisieren | Zeichenkette (1) | Nein | Optionale Angabe, ob bereits am Zählwerk zum Ablesezeitpunkt existierende Ablesungen aktualisiert werden sollen. Mögliche Werte: <ul style="list-style-type: none">• „J“ – ja, sollen aktualisiert werden• „N“ – nein, sollen nicht aktualisiert werden Default: „N“ |

Zählwerksstand

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|----------------|-------------------|-------------|---------------------------------|
| Zählwerknummer | Zeichenkette (16) | Ja/Nein (1) | Nummer des Zählwerkes am Zähler |

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|--------------|-----------------------|-------------|---|
| ObisCode | Zeichenkette (4) | Ja/Nein (1) | Obis-Code des Zählwerkes |
| Zählwerksart | Datum | Ja/Nein (1) | Zählwerksart des Abrechnungszählwerkes aus den Stammdaten der VA |
| Brennwert | Festkommazahl (12, 9) | Nein | Brennwert des übermittelten Standes Vorkommastellen: 3 Nachkommastellen: 9 |
| Zustandszahl | Festkommazahl (12, 9) | Nein | Zustandszahl des übermittelten Standes Vorkommastellen: 3 Nachkommastellen: 9 |
| Stand | Festkommazahl (18, 6) | Ja | Stand des Zählwerkes Vorkommastellen: 12 Nachkommastellen: 6 |

(1) Für die Ermittlung des korrekten Abrechnungszählwerkes ist eine der Angaben zu Zählwerksnummer, ObisCode oder Zählwerksart erforderlich.
Diese Angabe kann nur entfallen, wenn es sich um einen Zähler handelt, für den nur ein einzelnes Zählwerk existiert.

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde ein neuer Import angelegt
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für eine minimale Anfrage:

Nachfolgend ist ein Beispiel für die Übermittlung von Zählerständen zur Zählernummer EMH123456789 und dem Zählwerk mit dem OBIS-Code 1-1:1.8.0 im beschriebenen Format dargestellt:

```
Content-Type: application/sag.zaehlerstand.v1+json
[
    {
        "Zaehlernummer": "EMH123456789",
        "Geschaeftsbereich": "S",
        "Zeitpunkt": "2019-12-31T11:30+02:00",
        "Zaehlwerksstaende": [
            {
                "ObisCode": "1-1:1.8.0",
                "Stand": "100.000000"
            },
        ]
    }
]
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
    "Status": "404"
}
```

2.2.1.4 Änderung von Personen

Die Anfrage wird als PATCH Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Person/{PersonPin}>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Person/8711>

Wenn der Aufruf mit der Methode PATCH erfolgt und in der Anfrage ein JSON-formatiertes Objekt mit Bearbeiter-Informationen übergeben wird, so wird diese Information an den Personendatensatz geschrieben. Ist bereits ein Bearbeiter hinterlegt, so wird dieser überschrieben.

Die Grunddatenänderung der Person wird durchgeführt, wenn entsprechende Werte in den Elementen übergeben werden.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|---------------|--|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| AndernRequest | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja | JSON-formatierte Datenstruktur mit den Änderungs-Informationen |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

AndernRequest

Content-Type: application/sag.personaændern.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|-------------------|---------|---|
| Bearbeiter | Bearbeiter | Nein | JSON-formatierte Datenstruktur mit den Bearbeiter-Informationen |
| Name | Zeichenkette (70) | Nein | Name der Person |

| | | | |
|--------------|-------------------|------|--|
| Name2 | Zeichenkette (70) | Nein | Zweite Namensangabe bei juristischen Personen |
| Vorname | Zeichenkette (70) | Nein | Vorname der Person, nur bei natürlichen Personen Pflicht |
| Zusatz1 | Zeichenkette (70) | Nein | Zusatz 1 zum Namen |
| Zusatz2 | Zeichenkette (70) | Nein | Zusatz 2 zum Namen |
| Anrede | Zeichenkette (4) | Nein | <p>Angabe des Kürzels einer im Schleupen.CS System verwalteten Anrede. Folgende Werte sind bspw. möglich:</p> <ul style="list-style-type: none"> • Fr • Hr • Fam <p>Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten.</p> |
| Titel | Zeichenkette (8) | Nein | <p>Angabe des Kürzels eines im Schleupen.CS System verwalteten Titels. Folgende Werte sind bspw. möglich:</p> <ul style="list-style-type: none"> • Dr • ProfDr • RA <p>Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten.</p> |
| Geburtsdatum | Datum | Nein | <p>Angabe des Geburtsdatums bei natürlichen Personen</p> <p>Format: yyyy-MM-dd</p> <p>Beispiel: 2019-12-31</p> |

Bearbeiter

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|--------------------|---------|---|
| Id | Ganzzahl | Nein | Id des Sachbearbeiters aus, der für die Person zuständig sein soll. |
| Kuerzel | Zeichenkette (10) | Nein | Kürzel des Sachbearbeiters aus, der für die Person zuständig sein soll. |
| Email | Zeichenkette (120) | Nein | E-Mail Adresse des Sachbearbeiters aus, der für die Person zuständig sein soll. |

Es muss mindestens ein Element gefüllt sein, damit der Bearbeiter identifiziert werden kann.

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde ein Änderungsvorgang angelegt. Im Antwort-Kopf wird im Element „Location“ die relative Adresse zur Abfrage des Vorgangsstatus übermittelt (siehe Punkt 2.2.1.11) /Schleupen/IVP/SchnittstelleProzessanbindung/Vorgangsstatus/{Id_des_Vorgangs}
- 204 (NO CONTENT) - die Anfrage war erfolgreich, es wurden die Änderungen synchron ausgeführt
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben des Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für einen Bearbeiterzuordnung oder -änderung:

Nachfolgend ist ein Beispiel für die Übermittlung der Daten im beschriebenen Format dargestellt, um einen anderen Bearbeiter an einer Person zu hinterlegen:

```
Content-Type: application/sag.personaendern.v1+json
{
    "Bearbeiter": [
        {
            "Id": "53288",
            "Kuerzel": "ADM",
            "Email": "ADM@SCHLEUPEN.de"
        }
    ]
}
```

```
        "EMail": "admin@admin.de  
    }  
}
```

Beispieldatensatz für einen Personendatenänderung:

Nachfolgend ist ein Beispiel für die Übermittlung der Daten im beschriebenen Format dargestellt, um Daten einer Person zu ändern:

```
Content-Type: application/sag.personaendern.v1+json  
{  
    "Zusatz1": "von Hirschberg"  
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json  
{  
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder  
    Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem  
    aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",  
    "Status": "404"  
}
```

2.2.1.5 Änderung von Televerbindungen

Die Anfragen werden als POST, PUT und DELETE Methoden umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Person/{PIN_der_Person}/Televerbindung/{Id_der_Televerbindung}

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Person/6533/Televerbindung/821>

Diese Anfrage löscht in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ die Televerbindung mit dem Identifikator 821 der Person mit der PIN 6533, wenn der Aufruf als DELETE erfolgt. Erfolgt der Aufruf als PUT Methode und im Body der Anfrage werden entsprechende Daten übergeben, so wird die Televerbindung mit dem Identifikator 821 aktualisiert.

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Person/6533/Televerbindung/>

Diese Anfrage erzeugt eine neue Televerbindung an der Person mit der PIN 6533, wenn der Aufruf als POST Methode erfolgt und im Body der Anfrage entsprechende Daten übergeben werden.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|-----------------------|---|----------------------------------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| PIN der Person | Zeichenkette (in URI eingebettet) | Ja | Die PIN der Person, deren Auflistung von Televerbindungen geändert werden soll |
| Id der Televerbindung | Zeichenkette (in URI eingebettet) | Ja bei Löschung oder Änderung | Identifikator des Televerbindungsdatensatzes der Person, welche gelöscht oder geändert werden soll |

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|----------------|--|--------------------------------|--|
| Televerbindung | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja bei Neuanlage oder Änderung | JSON-formatierte Datenstruktur einer Televerbindung, welche angelegt oder geändert werden soll |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Televerbindung

Content-Type: application/sag.televerbindungsänderung.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|--------------------|---------|--|
| Art | Zeichenkette (8) | Ja | <p>Angabe der Art der Televerbindung als Kürzel laut Verwaltung der Stammdaten in Schleupen.CS. Mögliche Werte sind bspw.</p> <ul style="list-style-type: none"> • „Tel“ – Telefon • „e-mail“ – Email-Adresse • „mobil“ – Mobilfunknummer <p>Die vollständige Liste ist abhängig vom Customizing der jeweiligen Datenbank</p> |
| Nummer | Zeichenkette (120) | Ja | Nummer bzw. Adresse der Televerbindung, z.B. „987654321“, „0351/643222“ oder „max.mustermann@web.de“ |
| Vorwahl | Zeichenkette (50) | Nein | Optionale Angabe der Vorwahl zur Nummer, wenn diese getrennt vorliegt bzw. getrennt übernommen werden soll |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die POST-Anfrage war erfolgreich, es wurde die Televerbindung angelegt. Im http Header „Location“ der Antwortnachricht ist die relative Adresse zum Abruf, Ändern und Löschen der Televerbindung enthalten. Die relative Adresse ist nach folgendem Muster aufgebaut:

[/Schleupen/IVP/SchnittstelleProzessanbindung/Person/{PIN}/Televerbindung/{Id_der_Televerbindung}](#)

Da diese Schnittstelle den Abruf der Stammdaten nicht vorsieht, antwortet der GET-Service nur mit dem Status 204 (NO CONTENT).

- 204 (NO CONTENT) - die DELETE oder PUT-Anfrage war erfolgreich, die Televerbindung wurde gelöscht bzw.geändert

- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlercode und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlercode | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für die Anlage oder das Ändern einer Televerbindung:

```
Content-Type: application/sag.televerbindungsänderung.v1+json
{
    "Art": "Tel",
    "Vorwahl": "0815",
    "Nummer": "988743"
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
    "Fehlercode": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
    "Status": "404"
}
```

2.2.1.6 Änderung von Bankverbindungen

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Person/{PIN_der_Person}/Bankverbindung/

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Person/6533/Bankverbindung/>

Diese Anfrage erzeugt einen neuen Vorgang zur Änderung der Bankverbindung an der Person mit der PIN 6533. Im Body der Anfrage werden entsprechende Daten übergeben werden.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|------------------|---|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| PIN der Person | Zeichenkette (in URI eingebettet) | Ja | Die PIN der Person, deren Auflistung von Bankverbindungen geändert werden soll |
| Bankverbindungen | Liste von Objekten vom Typ Bankverbindung (im Nutzdatenbereich der Anfrage) | Ja | JSON-formatierte Datenstruktur einer oder mehrerer Bankverbindungen, welche angelegt oder geändert werden sollen |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Bankverbindung

Content-Type: application/sag.bankverbindungsaenderung.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|-------------------|---------|------------------------------------|
| IBAN | Zeichenkette (34) | Ja | Angabe der IBAN der Bankverbindung |
| Bankname | Zeichenkette (40) | Nein | Name der Bank |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|--------------------|-------------------|---------|---|
| BLZ | Zeichenkette (10) | Nein | Angabe der Bankleitzahl Wenn es sich nicht um eine deutsche IBAN handelt, muss die Bankleitzahl angegeben werden. |
| Kontonummer | Zeichenkette (20) | Nein | Angabe der Kontonummer Wenn es sich nicht um eine deutsche IBAN handelt, muss die Kontonummer angegeben werden. |
| BIC | Zeichenkette (11) | Nein | Angabe der BIC der Bank Wenn es sich nicht um eine deutsche IBAN handelt, muss die BIC angegeben werden |
| Inhaber | Zeichenkette (40) | Nein | Angabe des Inhabers der Bankverbindung |
| GueltigAb | Datum | Ja | Datum, ab wann die Bankverbindung gültig ist und somit verwendet werden kann Format: yyyy-MM-dd Beispiele: 2019-12-31 |
| GueltigBis | Datum | Nein | Datum, bis wann die Bankverbindung gültig ist und somit verwendet werden kann Format: yyyy-MM-dd Beispiele: 2019-12-31 |
| RechnungseinheitId | Ganzzahl | Nein | Angabe der Id der Rechnungseinheit, wenn die Bankverbindung für den Zahlungsverkehr dieser Rechnungseinheit genutzt werden soll |
| Art | Zeichenkette (2) | Nein | Mögliche Werte: FG – Forderungen und Guthaben FO – nur Forderungen GH – nur Guthaben Default: FG |
| Zweck | Zeichenkette (2) | Nein | Mögliche Werte: AR – Abschläge und Rechnungen AB – nur Abschläge RE – nur Rechnungen Default: AR |
| BriefErzeugen | Zeichenkette (1) | Nein | Gibt an, ob ein Brief zur Bankverbindungsänderung abgestellt werden soll. Wenn nicht angegeben, werden die Druckeinstellung aus CS.VA genutzt. Mögliche Werte: <ul style="list-style-type: none">• „J“ – Ja, Brief wird abgestellt• „N“ – Ja, Brief wird nicht abgestellt |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-----------------|-------------------|---------|--|
| BriefFormular | Zeichenkette (20) | Nein | Das Kürzel des zu verwendenden Briefformulars. Wenn nicht angegeben, werden die Druckeinstellung aus CS.VA genutzt. |
| BriefDruckDatum | Datum | Nein | Das gewünschte Druckdatum des Briefes. Muss in der Zukunft liegen. Format: yyyy-MM-dd Beispiele: 2019-12-31 |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde ein neuer Vorgang angelegt.
Im Antwort-Kopf wird im Element „Location“ die relative Adresse zur Abfrage des Vorgangsstatus übermittelt (siehe Punkt 2.2.1.11)
/Schleupen/IVP/SchnittstelleProzessanbindung/Vorgangsstatus/{Id_des_Vorgangs}
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlercode und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlercode | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für die Anlage einer Bankverbindung:

Das Beispiel erzeugt eine neue Bankverbindung an einer Person ohne Zuordnung zu einer Rechnungseinheit.

```
Content-Type: application/sag.bankverbindungsaenderung.v1+json
[
    {
        "IBAN": "DE66123456789012345678901234567890",
        "GueltigAb": "2021-06-25"
    }
]
```

Beispieldatensatz für die Anlage einer Bankverbindung mit Zuordnung zu einer Rechnungseinheit:

Das Beispiel erzeugt eine neue Bankverbindung an einer Person und ordnet diese der Rechnungseinheit zu. Wenn die Bankverbindung bereits existiert, wird nur die Zuordnung hergestellt.

```
Content-Type: application/sag.bankverbindungsaenderung.v1+json
[
  {
    "IBAN": "DE66123456789012345678901234567890",
    "GueltigAb": "2021-06-25",
    "RechnungseinheitId": "988743"
  }
]
```

Beispieldatensatz für das Beenden der Zuordnung zu einer Rechnungseinheit:

Das Beispiel beendet die Zuordnung einer Bankverbindung zu einer Rechnungseinheit. Die Bankverbindung selbst bleibt weiterhin gültig und kann an anderen Rechnungseinheiten weiterverwendet werden.

```
Content-Type: application/sag.bankverbindungsaenderung.v1+json
[
  {
    "IBAN": "DE66123456789012345678901234567890",
    "GueltigAb": "2021-06-25",
    "GueltigBis": "2021-06-31",
    "RechnungseinheitId": "988743"
  }
]
```

Beispieldatensatz für das Beenden einer Bankverbindung:

Das Beispiel beendet die Gültigkeit einer Bankverbindung. Alle bestehenden Zuordnungen zu Rechnungseinheiten werden ebenfalls beendet.

```
Content-Type: application/sag.bankverbindungsaenderung.v1+json
[
  {
    "IBAN": "DE66123456789012345678901234567890",
    "GueltigAb": "2021-06-25",
    "GueltigBis": "2021-06-31"
  }
]
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
  "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
  "Status": "404"
}
```

2.2.1.7 Änderung von Anschriften

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Person/{PIN_der_Person}/Anschrift/

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Person/6533/Anschrift>

Diese Anfrage legt in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ eine neue Anschrift zur Person mit der Pin 6533 an. Ein Beispiel für den Datencontainer wird im Konzept später dargestellt.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|----------------|--|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| PIN der Person | Ganzzahl (in URI eingebettet) | Ja | Die PIN der Person, deren Auflistung von Anschriften geändert werden soll |
| Anschrift | Anschrift (im Nutzdatenbereich der Anfrage) | Ja | JSON-formatierte Datenstruktur einer Anschrift, welche verarbeitet werden soll |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Anschrift

Content-Type: application/sag.anschriftsaenderung.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|-------------------|----------|---------------------------------|
| PLZ | Zeichenkette (10) | Ja | Postleitzahl der Anschrift |
| Ort | Zeichenkette (40) | Ja | Ortsangabe der Anschrift |
| Ortsteil | Zeichenkette (40) | Nein | Angabe des Ortsteiles |
| Strasse | Zeichenkette (40) | Nein (1) | Straßenangabe der Anschrift |
| Hausnummer | Zeichenkette (8) | Nein (1) | Hausnummernangabe der Anschrift |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------------|-------------------|----------|--|
| Hausnummernzusatz | Zeichenkette (4) | Nein | Der Zusatz zur Hausnummer |
| Hausnummernteil | Zeichenkette (8) | Nein | Zusätzliche Angabe zur Hausnummer |
| Postfach | Zeichenkette (10) | Nein (1) | Angabe des Postfachs |
| Land | Zeichenkette (4) | Nein | Landeskürzel der Anschrift (z.B. D für Deutschland) Standardwert: D |
| Art | Zeichenkette (4) | Nein | Angabe der Art der Anschrift als Kürzel laut Verwaltung der Stammdaten in Schleupen.CS. Mögliche Werte sind bspw. <ul style="list-style-type: none">• „AA“ – Allgemeine Anschrift• „RE“ – Rechnungsanschrift• „LI“ – Lieferanschrift Die vollständige Liste ist abhängig vom Customizing der jeweiligen Datenbank |
| Rechnungseinheit | Ganzzahl | Nein | Id einer Rechnungseinheit, welcher die Anschrift als Rechnungsanschrift zugeordnet werden soll. Diese Angabe ist optional, wenn nicht angegeben wird die Anschrift nur angelegt und nicht als Rechnungsanschrift zugeordnet. Wird eine Rechnungseinheit angegeben, so erfolgt die Verarbeitung asynchron über den Vorgangsdatenaustausch. Im anderen Fall wird die Anschrift direkt hinterlegt. Wird eine Anschrift an mehreren Rechnungseinheiten benötigt, so muss der Aufruf mehrfach mit den gleichen Anschriftendaten und unterschiedlichen Rechnungseinheiten erfolgt. Die Anschrift wird nicht doppelt angelegt, wenn diese bereits identisch an der Person hinterlegt ist. |

Hinweise:

(1) Anschriften können in drei verschiedenen Ausprägungen vorhanden sein:

- Normale Anschrift: Angabe von Straße und Hausnummer, Postfach ist nicht gefüllt
- Postfachanschrift: Angabe des Postfachs, Straße, Hausnummer und Hausnummernzusatz ist nicht gefüllt
- Großkundenanschrift: es ist weder Straße, Hausnummer und Hausnummernzusatz noch ein Postfach angegeben

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde die Anschrift angelegt. Im http Header „Location“ der Antwortnachricht ist die relative Adresse zum Abruf der Person enthalten. Die relative Adresse ist nach folgendem Muster aufgebaut: [/Schleupen/IVP/SchnittstelleProzessanbindung/Person/{PIN}](#)

Da diese Schnittstelle den Abruf der Stammdaten nicht vorsieht, antwortet der GET-Service nur mit dem Status 204 (NO CONTENT).

Wenn die Anschrift einer Rechnungseinheit zugeordnet werden soll, so erfolgt die Verarbeitung der Anfrage asynchron als Vorgang im Vorgangsdatenaustausch. Im Antwortkopf wird im Element „Location“ die relative Adresse zur Abfrage des Vorgangsstatus übermittelt (siehe Punkt 2.2.1.11)

[/Schleupen/IVP/SchnittstelleProzessanbindung/Vorgangsstatus/{Id_des_Vorgangs}](#)

- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlercode und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben des Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlercode | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für die minimalste Anschriftenänderung:

Nachfolgend ist ein Beispiel für die Anlage einer Großkundenanschrift. Dieser Datencontainer stellt den minimalsten möglichen Umfang dar.

```
Content-Type: application/sag.anschriftsaenderung.v1+json
{
    "PLZ": "01067",
    "Ort": "Dresden"
}
```

Beispieldatensatz für die Anlage einer normalen Anschrift:

Nachfolgend ist ein Beispiel für die Anlage normalen Anschrift mit Straße und Hausnummer.

```
Content-Type: application/sag.anschriftsaenderung.v1+json
{
    "PLZ": "01067",
    "Ort": "Musterstadt",
    "Strasse": "Musterstrasse",
    "Hausnummer": "1",
    "Art": "AA"
}
```

Beispieldatensatz für die Anlage einer normalen Anschrift:

Nachfolgend ist ein Beispiel für die Anlage normalen Anschrift mit Straße und Hausnummer sowie der Zuordnung der Anschrift zu einer Rechnungseinheit als Rechnungsanschrift.

```
Content-Type: application/sag.anorschiftsaenderung.v1+json
{
    "PLZ": "01067",
    "Ort": "Musterstadt",
    "Strasse": "Musterstrasse",
    "Hausnummer": "1",
    "Art": "AA",
    "Rechnungseinheit": 976,
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
    "Status": "404"
}
```

2.2.1.8 Änderung von Abschlägen

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben.

Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Abschlagsaenderung>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden.

Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Abschlagsaenderung>

Diese Anfrage erzeugt einen neuen Vorgang zur Änderung des Abschlags eines Vertrags in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“.

Request

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------------------------------|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der zu verwendenden Datenquelle aus Schleupen.CS |

| | | | |
|--------------------|--|----|--|
| Mandant | Zeichenkette (8) (Header) | Ja | Angabe des Kürzels des zu verwendenden Mandanten aus Schleupen.CS |
| Abschlagsaenderung | Objekt vom Typ Abschlagsaenderung (im Nutzdatenbereich der Anfrage) | Ja | Fachliche Daten als JSON-formatiertes Objekt im Nutzdatenbereich der Anfrage |

Abschlagsaenderung

Content-Type: application/sag.abschlagsaenderung.v1+json

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|---------------------------|-------------------|---------|--|
| Vertragsnummer | Zeichenkette (16) | Ja | Die Angabe der eindeutigen Vertragsnummer des Vertrages, dessen Abschlag geändert werden soll |
| Geschaeftsbereich | Zeichenkette (4) | Ja | Kürzel des Geschäftsbereiches des Vertrags |
| Abschlag | Festkommazahl | Ja | Angabe der Höhe des Abschlags |
| GueltigAb | Datum | Ja | Angabe des Datums, ab wann der neue Abschlagsbetrag gültig sein soll Format: yyyy-MM-dd Beispiele: 2019-12-31 |
| Anzahl | Ganzzahl | Nein | Optionale Angabe der Anzahl der Abschläge pro Jahr, wenn diese z.B. im Zuge der Änderung der nächsten Sollstellung in Verbindung mit dem Abschlagsplan angepasst werden sollen. |
| DatumNaechsteSollstellung | Datum | Nein | Optionale Angabe des Datums zur Durchführung der nächsten Sollstellung für den Vertrag. Das Datum stellt das früheste Datum dar und wird ggf. durch den Vorgangsaustausch laut Abschlagsplan korrigiert. Wird kein Wert übergeben, so ermittelt die Vorgangsverarbeitung, falls notwendig, den neuen Wert. Format: yyyy-MM-dd Beispiele: 2019-12-31 |
| Ermittlungsart | Aufzählung | Nein | Optionale Angabe der Ermittlungsart des Abschlags, falls diese geändert werden soll. Mögliche Werte sind: <ul style="list-style-type: none"> • FesterBetragProAbrechnungsperiode • ImmerAendernWennNiedrigerAlsAlt |

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|--------------------|----------------------|---------|---|
| | | | <ul style="list-style-type: none"> • FesterBetrag • GleichmaessigLetzterAbschlagMitRest • EinmaligAendernWennHoherAlsAlt • ImmerAendernWennHoherAlsAlt • Keine • EinmaligAendernWennNiedrigerAlsAlt • Berechnen |
| BriefdruckDatum | Datum | Nein | <p>Optionale Angabe des Briefdruckdatums für den Kundenbrief zur Information über die durchgeführte Abschlagsänderung. Wenn kein Datum angegeben wird, wird kein Brief abgestellt.</p> <p>Format: yyyy-MM-dd</p> <p>Beispiele: 2019-12-31</p> |
| BriefdruckFormular | Zeichenkette | Nein | Optionale Angabe des Kürzels des Briefformulars für die Erzeugung des Abschlagsbriefes. Hier muss immer dann eine Angabe vorgenommen werden, wenn es mehr als ein mögliches Formular für die Erzeugung des Abschlagsbriefes im System gibt und keine eindeutige Auswahl durch den Vorgangsverarbeitung getroffen werden kann. |
| Herkunft | Zeichenkette (35) | Nein | Optionale Angabe der Herkunft zur Identifikation der Vorgänge, welche aus dem Fremdsystem übernommen wurden. Der Standardwert ist „PORTAL“. |
| HerkunftVersion | Zeichenkette (10) | Nein | Optionale Angabe der Version zur Identifikation der Vorgänge, welche aus dem Fremdsystem übernommen wurden. Der Standardwert ist „1.0“. |
| Identifikation | Zeichenkette (36) | Nein | <p>Optionale Vorgabe eines Identifikationswertes für den Vorgang der Abschlagsänderung in Form eines 36-stelligen Guid (bspw. ecfecf73-7ebf-4d07-b80f-383dff4b154d).</p> <p>Wird kein Wert vorgegeben, so wird dieser automatisch generiert. Anhand des Identifikationswertes kann der Verarbeitungsstatus des asynchronen Vorgangs der Abschlagsänderung abgerufen werden.</p> |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde ein neuer Vorgang angelegt.
Im Antwort-Kopf wird im Element „Location“ die relative Adresse zur Abfrage des Vorgangsstatus übermittelt (siehe Punkt 2.2.1.11)
[/Schleupen/IVP/SchnittstelleProzessanbindung/Vorgangsstatus/{Id_des_Vorgangs}](#)
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlercode und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlercode | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

2.2.1.9 Änderung von Rechnungseinheiten

Die Anfragen sind als POST, PUT und DELETE Methoden umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Rechnungseinheit/{id}/{action}>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden.
Folgende actions werden unterstützt:

- (PUT) Notiz
- (PUT) Maßnahme
- (DELETE) Maßnahme
- (POST) Rechnungsempfänger

Mögliche Abfragen können beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Rechnungseinheit/5864/Notiz>

Diese Anfrage fügt in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ der Rechnungseinheit mit der Id 5864 eine Notiz hinzu, wenn der Aufruf als PUT erfolgt und im Body der Anfrage werden entsprechende Daten übergeben wurden.

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Rechnungseinheit/5864/Maßnahme>

Diese Anfrage fügt in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ der Rechnungseinheit mit der Id 5864 eine Maßnahme hinzu, wenn der Aufruf als PUT erfolgt und im Body der Anfrage werden entsprechende Daten übergeben wurden. Wird der Aufruf als DELETE durchgeführt, wird die Maßnahme der Rechnungseinheit wieder entfernt.

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Rechnungseinheit/5864/Rechnungsempfaenger>

Diese Anfrage ändert in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ an der Rechnungseinheit mit der Id 5864 den Rechnungsempfänger, wenn der Aufruf als PUT erfolgt und im Body der Anfrage werden entsprechende Daten übergeben wurden.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|-------------|---|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|-------------------------|--|--|---|
| Id der Rechnungseinheit | Zeichenkette (in URI eingebettet) | Ja | Die Datenbank Id der Rechnungseinheit, welche geändert werden soll |
| Notiz | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja bei Neuanlage oder Änderung der Notiz | JSON-formatierte Datenstruktur einer Notiz, welche angelegt oder geändert werden soll |
| Massnahme | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja bei Neuanlage oder Änderung der Massnahme | JSON-formatierte Datenstruktur einer Massnahme, welche angelegt oder geändert werden soll |
| Rechnungsempfänger | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja bei Neuanlage oder Änderung des Rechnungsempfängers | JSON-formatierte Datenstruktur eines Rechnungsempfängers, welche angelegt oder geändert werden soll |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Notiz

Content-Type: application/sag.notiz.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|--------------|---------|----------------|
| Text | Zeichenkette | Ja | Text der Notiz |

Massnahme

Content-Type: application/sag.massnahme.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|----------------------|---------|---|
| Datum | Datum mit Zeitangabe | Ja | Zeitpunkt der Maßnahme |
| Kuerzel | Zeichenkette (4) | Ja | Kürzel der Maßnahme, worüber diese dann ermittelt und an der Rechnungseinheit hinterlegt wird. Die Maßnahmen können in der Finanzbuchhaltung unter Stammdaten > Grunddaten > Maßnahmen verwaltet werden. |

Rechnungsempfänger

Content-Type: application/sag.rechnungsempfaenger.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|----------|-------------|--------------------|
| PIN | Ganzzahl | Ja/Nein (1) | Die PIN der Person |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|--------------|--------------------------|----------------|--|
| Art | Auflistung | Ja/Nein (1) | Angabe, ob es sich um eine natürliche oder juristische Person handelt. Mögliche Werte: <ul style="list-style-type: none">• „J“ – juristische Person• „N“ – natürliche Person |
| Name | Zeichenkette (70) | Ja/Nein (1) | Name der Person |
| Name2 | Zeichenkette (70) | Nein | Zweite Namensangabe bei juristischen Personen |
| Vorname | Zeichenkette (70) | Nein | Vorname der Person, nur bei natürlichen Personen Pflicht |
| Zusatz1 | Zeichenkette (70) | Nein | Zusatz 1 zum Namen |
| Zusatz2 | Zeichenkette (70) | Nein | Zusatz 2 zum Namen |
| Anrede | Zeichenkette (4) | Nein | Angabe des Kürzels einer im Schleupen.CS System verwalteten Anrede. Folgende Werte sind bspw. möglich: <ul style="list-style-type: none">• Fr• Hr• Fam Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten. |
| Titel | Zeichenkette (8) | Nein | Angabe des Kürzels eines im Schleupen.CS System verwalteten Titels. Folgende Werte sind bspw. möglich: <ul style="list-style-type: none">• Dr• ProfDr• RA Der konkrete Wert ist abhängig vom Customizing und somit von den in Schleupen.CS verwalteten Werten. |
| Geburtsdatum | Datum | Nein | Angabe des Geburtsdatums bei natürlichen Personen Format: yyyy-MM-dd Beispiel: 2019-12-31 |
| Anschrift | Objekt vom Typ Anschrift | Nein | Anschrift des Rechnungsempfängers |
| Bearbeiter | Zeichenkette (10) | Nein | Kürzel vom Benutzer des Sachbearbeiters für die Person |

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|---------------|---------|---|
| DSGVOOptIn | Wahrheitswert | Nein | <p>Kennzeichen für die vom Kunden erteilte/nicht erteilte Werbeerlaubnis</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • true • false |

- (1) Bei der Übergabe der Person kann entweder eine bereits im System vorhandene Person referenziert werden oder es werden die vollständigen Fachdaten für die neu anzulegende Person übergeben. Im ersten Fall wird nur die PIN übermittelt, im zweiten Fall wird keine PIN angegeben und stattdessen mindestens Art und Name

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die POST-Anfrage war erfolgreich, es wurde die Änderung der an der Rechnungseinheit durchgeführt. Im http Header „Location“ der Antwortnachricht ist die relative Adresse zum Abruf, Ändern der Notiz, Maßnahme oder des Rechnungsempfängers einer Rechnungseinheit enthalten. Die relative Adresse ist nach folgendem Muster aufgebaut:
</Schleupen/IVP/SchnittstelleProzessanbindung/> [Rechnungseinheit/{id}/{action}](#)
- Da diese Schnittstelle den Abruf der Stammdaten nicht vorsieht, antwortet der GET-Service nur mit dem Status 204 (NO CONTENT).
- 204 (NO CONTENT) - die DELETE oder PUT-Anfrage war erfolgreich, die Maßnahme der Rechnungseinheit wurde gelöscht-
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlercode und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben des Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für die Anlage oder Änderung einer Notiz an einer Rechnungseinheit:

Content-Type: application/sag.personaendern.v1+json
{
 "Text": "Hier steht die Notiz dieser Rechnungseinheit!"
}

Beispieldatensatz für das Setzen einer Maßnahme einer Rechnungseinheit:

Content-Type: application/sag.personaendern.v1+json
{
 "isSet": true,
 "note": "Testnotiz",
 "id": 10,
 "employee": "PL",
 "billingUnitId": "a005r000002B8ftAAC",
 "actionDate": "2022-06-22"
}

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

Content-Type: application/sag.fehlermeldung.v1+json
{
 "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
 "Status": "404"
}

2.2.1.10 Verwalten von Eigentümern an Abnahmestellen

Die Anfrage wird als POST und DELETE Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Abnahmestelle/{Id_der_Abnahmestelle}/Eigentuemer

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Abnahmestelle/8711/Eigentuemer>

Wird diese Adresse mit der Methode DELETE aufgerufen, so wird ein ggf. vorhandener Eigentümer an der Abnahmestelle 8711 entfernt. Wenn der Aufruf mit der Methode POST erfolgt und in der Anfrage ein JSON-formatiertes Objekt mit Eigentümerinformationen übergeben wird, so wird diese Information an die Abnahmestelle geschrieben. Ist bereits ein Eigentümer hinterlegt, so wird dieser überschrieben.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|-------------------------|--|---|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| Id der Abnahmestelle | Ganzzahl (in Uri eingebettet) | Ja | Technischer Schlüssel der Abnahmestelle, an welcher der Eigentümer ergänzt, aktualisiert oder entfernt werden soll |
| Eigentuemer | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja nur bei Anlage oder Änderung | JSON-formatierte Datenstruktur mit den Eigentümerinformationen (nur bei Aufruf per POST) |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Eigentuemer

Content-Type: application/sag.eigentuemer.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------|----------|---------|--|
| PIN | Ganzzahl | Ja | Die PIN der Person, welche als Eigentümer hinterlegt werden soll. Die Person muss im Vorfeld bereits anderweitig angelegt wurden sein (z.B. durch einen Aufruf der Anlage von Personen, siehe Punkt 2.2.1.1) |
| AnschriftId | Ganzzahl | Ja | Der technische Schlüssel einer Anschrift der Person, welche Eigentümer werden soll. Diese Anschrift wird als Eigentümeranschrift der Abnahmestelle zugeordnet |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde der Eigentümer mit seiner Anschrift an der Abnahmestelle hinterlegt
- 204 (NO CONTENT) - die DELETE-Anfrage war erfolgreich, die Eigentümerinformationen wurden entfernt
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlercode und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlercode | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für einen Eigentümerzuordnung oder Änderung:

Nachfolgend ist ein Beispiel für die Übermittlung der Daten im beschriebenen Format dargestellt, um einen neuen Eigentümer an einer Abnahmestelle zu hinterlegen:

Content-Type: application/sag.eigentuemer.v1+json

```
{  
    "PIN": "53288",  
    "AnschriftId": 75674  
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder
Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem
aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
    "Status": "404"
}
```

2.2.1.11 Abfrage des Vorgangsstatus

Die Anfrage wird als GET Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse und in den Kopfinformationen der Anfrage übergeben.

Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Vorgangsstatus/{Id_des_Vorgangs}

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden.

Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Vorgangsstatus/ecfecf73-7ebf-4d07-b80f-383dff4b154d>

Diese Anfrage ruft in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ den Status des Vorgangs mit der Id ecfecf73-7ebf-4d07-b80f-383dff4b154d ab.

Die Suche nach dem Vorgang erfolgt anhand des übermittelten Identifikators. Der Vorgang muss im Vorfeld durch einen anderen Webservice entsprechend angelegt worden sein.

Request

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-----------------|---|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der zu verwendenden Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (8) (Header) | Ja | Angabe des Kürzels des zu verwendenden Mandanten aus Schleupen.CS |
| Id des Vorgangs | Zeichenkette (in URI eingebettet) | Ja | Der Schlüsselwert des angelegten Vorgangs, dessen Status abgerufen werden soll |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 200 (OK) - die Anfrage war erfolgreich, es werden die Statusinformationen des Vorgangs als in JSON formatierte Daten an den Aufrufer übertragen
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlercode und ggf. die Protokolle des Servers analysiert werden.

Beschreibung der Ergebnisobjekte

Vorgangsstatus

Content-Type: application/sag.vorgangsstatus.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------------|--------------|---------|---|
| Id | Zeichenkette | Ja | Schlüsselwert des angefragten Vorgangs im Schleupen.CS System. |
| Status | Zeichenkette | Ja | Kürzel des Verarbeitungsstatus des Vorgangs. Mögliche Werte sind: <ul style="list-style-type: none">• IS - Vorgang wird importiert.• IW - Vorgang wartet auf Verarbeitung.• VW - Vorgang in Bearbeitung.• VE - Vorgang fehlerhaft verarbeitet.• VM - Vorgang fehlerhaft verarbeitet, aber manuell umsetzbar.• VI - Vorgang erfolgreich verarbeitet.• GM - wie VM und zusätzlich Fehler bei Benachrichtigung des Geschäftsprozesses• GE - wie VE und zusätzlich Fehler bei Benachrichtigung des Geschäftsprozesses• GI - wie VI und zusätzlich Fehler bei Benachrichtigung des Geschäftsprozesses |
| Statusbezeichnung | Zeichenkette | Ja | Textuelle Beschreibung des Statuskürzels |
| Fehler | Zeichenkette | Nein | Optionale Angabe eines Fehlertextes, wenn der Vorgang nicht erfolgreich verarbeitet werden konnte |

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben des Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für einen erfolgreichen Abruf:

Nachfolgend ist ein Beispiel für die Übermittlung der Statusinformationen eines Vorgangs dargestellt:

Content-Type: application/sag.vorgangssstatus.v1+json

```
{  
    "Id": "ecfecf73-7ebf-4d07-b80f-383dff4b154d",  
    "Status": "VI",  
    "Statusbezeichnung": "Vorgang erfolgreich verarbeitet"  
}
```

Beispieldatensatz für einen fehlgeschlagenen Abruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

Content-Type: application/sag.fehlermeldung.v1+json

```
{  
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder  
    Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem  
    aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",  
    "Status": "404"  
}
```

2.2.1.12 Abfrage des Geschäftsprozessstatus

Die Anfrage wird als GET Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse und in den Kopfinformationen der Anfrage übergeben.

Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Geschaeftsprozessstatus/{Id_des_Geschaeftsprozesses}

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden.

Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Geschaeftsprozessstatus/8711>

Diese Anfrage ruft in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“ den Status zum Geschäftsprozess mit der Id 8711 ab.

Die Suche nach dem Geschäftsprozess erfolgt anhand des übermittelten Identifikators. Der Prozess muss im Vorfeld durch einen anderen Webservice entsprechend angelegt worden sein (z.B. Vertragsanlage oder Vertragsbeendigung).

Request

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------------------------|---|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der zu verwendenden Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (8) (Header) | Ja | Angabe des Kürzels des zu verwendenden Mandanten aus Schleupen.CS |
| Id des Geschaeftsprozesses | Zeichenkette (in URI eingebettet) | Ja | Der Schlüsselwert des angelegten Geschäftsprozesses, dessen Status abgerufen werden soll |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 200 (OK) - die Anfrage war erfolgreich, es werden die Statusinformationen des Geschäftsprozesses als in JSON formatierte Daten an den Aufrufer übertragen
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)

- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Beschreibung der Ergebnisobjekte

Geschäftsprozessstatus

Content-Type: application/sag.geschaeftsprozessstatus.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|----------------|---|---------|---|
| Id | Ganzzahl | Ja | Schlüsselwert des angefragten Geschäftsprozesses im Schleupen.CS System. |
| Status | Zeichenkette | Ja | Textuelle Beschreibung des Status, z.B. „Warte Beendig. Marktk. (S001)“ |
| Beschreibung | Zeichenkette | Nein | Nähere Beschreibung zum Status, z.B. „Warten auf Beendigung der Marktkommunikation.“ |
| Fehler | Zeichenkette | Nein | Optionale Angabe eines Fehlertextes, wenn der Prozess nicht erfolgreich verarbeitet werden konnte oder wenn in untergeordneten Vorgängen ein Fehler oder eine Warnung aufgetreten ist. Das Auftreten eines Fehlertextes bedeutet nicht, dass der Geschäftsprozess insgesamt fehlerhaft abgebrochen wurde. |
| Vorgangsstatus | Zeichenkette | Nein | Aktueller Status des angelegten VDM-Vorgangs in Schleupen.CS. Mögliche Werte: <ul style="list-style-type: none"> • Aktiv • ErfolgreichBeendet • FehlerhaftBeendet |
| Zeitpunkt | Datum mit Zeitangabe | Nein | Zeitpunkt, an welchem der Status zuletzt geändert wurde Format: yyyy-MM-ddThh:mm:ssTZD Beispiele: 2019-12-31T11:30:15+02:00 |
| Vertragsnummer | Zeichenkette | Nein | Angabe der Vertragsnummer, wenn es sich um einen Vertragsanlageprozess handelt und dieser erfolgreich abgeschlossen wurde |
| Marktmeldungen | Liste von Objekten vom Typ StatusMarktmeldung | Nein | Liefert die Statusinformationen zu den Marktmeldungen, welche mit dem Prozess verknüpft sind (z.B. Anmeldung Netznutzung oder Kündigung Vorlieferant) |

StatusMarktmeldung

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|-------------------|--|---------|---|
| Prozessnummer | Ganzzahl | Nein | Prozessnummer in Schleupen.CS |
| Vorgangsnummer | Ganzzahl | Nein | Vorgangsnummer in Schleupen.CS |
| Meldungsnummer | Ganzzahl | Nein | Meldungsnummer in Schleupen.CS |
| Meldungsreferenz | Zeichenkette | Nein | Referenz der Meldung, wenn bereits erzeugt |
| Kategorie | Zeichenkette | Nein | Kategorie der Meldung laut UTILMD Spezifikation |
| Transaktionsgrund | Zeichenkette | Nein | Transaktionsgrund der Meldung laut UTILMD Spezifikation |
| Status | Zeichenkette | Nein | Meldungsstatus |
| Fehlertext | Zeichenkette | Nein | Beschreibung eines Fehlers |
| Antworten | Liste von Objekten vom Typ AntwortMarktmeldung | Nein | Antworten der Marktkommunikation zu dieser Meldung |

AntwortMarktmeldung

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|--------------------|----------------------|---------|---|
| Timestamp | Datum mit Zeitangabe | Nein | Zeitpunkt |
| Prozess | Zeichenkette (50) | Nein | Kategorie-Kürzel und Beschreibung der Marktmeldung |
| AntwortKuerzel | Zeichenkette (7) | Nein | Kürzel des Antworttyps Bsp.: „E14“ |
| AntwortBezeichnung | Zeichenkette () | Nein | Bezeichnung des Antworttyps Bsp.: „Ablehnung Sonstiges“ |
| AntwortPositiv | Zeichenkette (1) | Ja | Gibt an ob die Rückmeldung zu diesem Prozess positiv oder negativ ist. Mögliche Auswahlen: <ul style="list-style-type: none">• „J“ – positiv• „N“ – negativ |

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben des Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für einen erfolgreichen Abruf:

Nachfolgend ist ein Beispiel für die Übermittlung der Statusinformationen eines Vorgangs dargestellt:

```
Content-Type: application/sag.geschaeftsprozessstatus.v1+json
{
    "Id": 8711,
    "Status": "Warte Beendig. Marktk. (S001)",
    "Beschreibung": "Warten auf Beendigung der Marktkommunikation.",
    "Fehler": "DOMVAIMP013 Geräteeinbau - Das Datum liegt zu weit in der Zukunft! Hinweis zu Beginndatum Gerät: 23445576<n>DOMVAIMP012 Vertrag: Fehler beim Ermitteln der Abschläge.&lt;n>DOMVAIMP012 Vertrag: Fehler in Abschlagsberechnung. TB auf 0 gesetzt. Prüfen!",
    "Vorgangsstatus": "Aktiv",
    "Zeitpunkt": "2021-02-17T16:57:52+01:00",
    "Marktmeldungen": [
        {
            "Prozessnummer": 1755,
            "Vorgangsnummer": 2725,
            "Meldungsnummer": 2290,
            "Kategorie": "E01",
            "Transaktionsgrund": "E03",
            "Status": "mp",
        }
    ]
}
```

Beispieldatensatz für einen fehlgeschlagenen Abruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
    "Status": "404"
}
```

2.2.1.13 Wechsel von Tarifen

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Tarifwechsel>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Tarifwechsel>

Diese Anfrage erzeugt einen neuen Vorgang zur Änderung des Tarifs eines Vertrags in der Schleupen.CS Datenquelle „Echtdatenbank“ im Mandanten „0001“.

Request

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|--------------|---|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der zu verwendenden Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (8) (Header) | Ja | Angabe des Kürzels des zu verwendenden Mandanten aus Schleupen.CS |
| Tarifwechsel | Objekt vom Typ Tarifwechsel (im Nutzdatenbereich der Anfrage) | Ja | Fachliche Daten als JSON-formatiertes Objekt im Nutzdatenbereich der Anfrage |

Tarifwechsel

Content-Type: application/sag.tarifwechsel.v1+json

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|----------------------|--|---------|---|
| Vertragsnummer | Zeichenkette (16) | Ja | Die Angabe der eindeutigen Vertragsnummer des Vertrages, dessen Abschlag geändert werden soll |
| Geschaeftsbereich | Zeichenkette (4) | Ja | Kürzel des Geschäftsbereiches des Vertrags |
| GueltigAb | Datum | Ja | Angabe des Datums, ab wann der neue Abschlagsbetrag gültig sein soll Format: yyyy-MM-dd Beispiele: 2019-12-31 |
| Tarifkuerzel | Zeichenkette (8) | Ja | Kürzel des neuen Tarifs, welcher für den Vertrag genutzt werden soll |
| Abrechnungsfeldwerte | Liste von Objekten vom Typ Abrechnungsfeldwert | Nein | Optionale Angabe von Abrechnungsfeldwerten, welche an die neue Zeitscheibe mit dem neuen Tarif übernommen werden sollen |

Abrechnungsfeldwert

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|--------------|-------------------|---------|--|
| Feldreferenz | Zeichenkette (10) | Ja | Feldreferenz, welche das Abrechnungsfeld des Tarifs identifiziert, für welches ein Wert an den Vertrag gesetzt werden soll |
| Wert | Zeichenkette (40) | Ja | Wert, welcher für dieses Abrechnungsfeld am Vertrag gesetzt werden soll |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde der Prozess für den Tarifwechsel angelegt. Im Antwort-Kopf wird im Element „Location“ die relative Adresse zur Abfrage des Prozessstatus übermittelt (siehe Punkt 2.2.1.12) [/Schleupen/IVP/SchnittstelleProzessanbindung/Geschaeftsprozesstatus/{Id_des_Geschaeftsprozesses}](#)
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für einen Tarifwechselprozess:

Nachfolgend ist ein Beispiel für die Übermittlung der Daten im beschriebenen Format dargestellt, um einen neuen Prozess für den Tarifwechsel zum 01.01.2022 auszulösen:

Content-Type: application/sag.tarifwechsel.v1+json

```
{  
    "Vertragsnummer": "53288",  
    "Geschaeftsbereich": "S",  
    "Tarifkuerzel": "FIX24",  
    "GueltigAb": "2022-01-01",  
    "Abrechnungsfeldwerte": [  
        {  
            "Feldreferenz": "VP.Schlüssel",  
            "Wert": "01"  
        }  
    ]  
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

Content-Type: application/sag.fehlernmeldung.v1+json

```
{  
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder  
    Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem  
    aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",  
    "Status": "404"  
}
```

2.2.1.14 Auflösung von Verträgen

Die Anfrage wird als POST Methode umgesetzt. Die Anfrageparameter werden in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Vertragsauflösung>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://csappserver/Schleupen/IVP/SchnittstelleProzessanbindung/Vertragsauflösung>

Diese Anfrage erzeugt einen neuen Prozess zur Vertragsauflösung. Im Body der Anfrage werden entsprechende Daten übergeben werden.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|-------------------|--|---------|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| Vertragsauflösung | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja | JSON-formatierte Datenstruktur einer Vertragsauflösung, für welche ein Prozess angelegt werden soll |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Vertragsauflöesung

Content-Type: application/sag.vertragsauflöesung.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|------------------|-------------------|---------|--|
| Vertragsnummer | Zeichenkette (16) | Ja | Die Nummer des aufzulösenden Vertrags |
| Geschäftsbereich | Zeichenkette (4) | Ja | Kürzel des Geschäftsbereiches des Vertrags |
| Vertragsende | Datum | Ja | Das Datum, zu welchem der Vertrag aufgelöst werden soll Format: yyyy-MM-dd Beispiele: 2019-12-31 |

| | | | |
|-------------------|-------------------|------|---|
| Transaktionsgrund | Zeichenkette (3) | Nein | Angabe eines gültigen Kürzels laut Definition UTILMD für den Transaktionsgrund, welcher in der Marktkommunikation genutzt wird Wenn kein Wert übergeben wird, so wird E01 genutzt. |
| Herkunft | Zeichenkette (35) | Nein | Optionale Angabe der Herkunft zur Identifikation der Vorgänge, welche aus dem Fremdsystem übernommen wurden. Der Standardwert ist „PORTAL“. |
| HerkunftVersion | Zeichenkette (10) | Nein | Optionale Angabe der Version zur Identifikation der Vorgänge, welche aus dem Fremdsystem übernommen wurden. Der Standardwert ist „1.0“. |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde der Prozess für die Vertragsauflösung angelegt. Im Antwort-Kopf wird im Element „Location“ die relative Adresse zur Abfrage des Prozessstatus übermittelt (siehe Punkt 2.2.1.12) [/Schleupen/IVP/SchnittstelleProzessanbindung/Geschaeftsprozesstatus/{Id_des_Geschaeftsprozesses}](#)
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben des Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für einen Vertragsauflösungsprozess:

Nachfolgend ist ein Beispiel für die Übermittlung der Daten im beschriebenen Format dargestellt, um einen neuen Prozess für die Vertragsauflösung auszulösen:

```
Content-Type: application/sag.vertragsaufloesung.v1+json
{
    "Vertragsnummer": "53288",
    "Vertragsende": "2021-12-31",
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

```
Content-Type: application/sag.fehlermeldung.v1+json
{
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",
    "Status": "404"
}
```

2.2.1.15 Setzen von Zahlsperren

Die Anfrage wird als PUT und DELETE Methode umgesetzt. Die Anfrageparameter werden in der aufgerufenen Adresse, in den Kopfinformationen und als JSON-formatiertes Datenobjekt im Body der Anfrage übergeben. Die Aufrufadresse wird nach folgendem Muster zusammengebaut:

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Buchung/{id}/Zahlsperre>

Die Elemente in geschweiften Klammern müssen mit konkreten Werten ersetzt werden. Nachfolgend werden diese Parameter tabellarisch dargestellt. Eine mögliche Abfrage kann beispielsweise folgenden Aufbau haben:

HTTP Header:

- X-Datasource: Echtdatenbank
- X-Mandant: 0001

<http://{Servername}/Schleupen/IVP/SchnittstelleProzessanbindung/Buchung/3865/Zahlsperre>

Wird diese Adresse mit der Methode DELETE aufgerufen, so wird ein ggf. vorhandene Zahlsperre der Buchung 3865 entfernt. Wenn der Aufruf mit der Methode PUT erfolgt und in der Anfrage ein JSON-formatiertes Objekt mit Zahlsperren-Informationen übergeben wird, so wird diese Information an die Buchung geschrieben, bzw. die Zahlsperre gesetzt. Ist bereits eine Zahlsperre hinterlegt, so wird dieser überschrieben.

Request

| Eigenschaft | Datentyp | Pflicht | Beschreibung |
|-------------|--|---|--|
| Servername | Zeichenkette (in URI eingebettet) | Ja | Die Angabe des Namens oder der IP Adresse des Servers, auf welchem die Schnittstelle installiert ist |
| Datenquelle | Zeichenkette (Header) | Ja | Angabe der verwendeten Datenquelle aus Schleupen.CS |
| Mandant | Zeichenkette (Header) | Ja | Angabe des zu verwendenden Mandanten aus Schleupen.CS |
| Zahlsperre | Datenstruktur (im Nutzdatenbereich der Anfrage) | Ja nur bei Anlage oder Änderung | JSON-formatierte Datenstruktur mit den Zahlsperren-Informationen (nur bei Aufruf per POST) |

Nachfolgend ist tabellarisch der Aufbau des Nutzdatencontainers beschrieben. Die Übertragung erfolgt im JSON-Format.

Zahlsperre

Content-Type: application/sag.zahlsperre.v1+json

| Bezeichnung | Datentyp | Pflicht | Beschreibung |
|---------------|----------------------|---------|--|
| ZahlsperreBis | Datum mit Zeitangabe | Nein | Zeitpunkt, wann die Zahlsperre automatisch aufgehoben werden soll. |
| Grund | Zeichenkette | Nein | Begründung für das setzen der Zahlsperre |

Response

Die Anfrage kann mit folgenden Statuscodes beantwortet werden:

- 201 (CREATED) - die Anfrage war erfolgreich, es wurde die Zahlsperre gesetzt
- 204 (NO CONTENT) - die DELETE-Anfrage war erfolgreich, die Zahlsperre wurde entfernt
- 400 (BAD REQUEST) - die Anfrage ist ungültig. Es wurden beispielsweise nicht unterstützte Parameter genutzt
- 404 (NOT FOUND) - die Anfrage ist syntaktisch korrekt, es wurde jedoch falsche Parameterwerte übergeben (z.B. die Datenquelle oder der Mandant existieren nicht)
- 500 (INTERNAL SERVER ERROR) - es ist ein genereller Fehler bei der Verarbeitung der Anfrage aufgetreten. In diesen Fällen müssen der gelieferte Fehlertext und ggf. die Protokolle des Servers analysiert werden.

Fehlerobjekt

Eine fehlerhaft verarbeitete Anfrage liefert neben dem Statuscode im http Header ein Fehlerobjekt in JSON Formatierung mit nachfolgenden Elementen.

| Eigenschaft | Datentyp | Pflicht | Bemerkung |
|-------------|--------------|---------|--|
| Fehlertext | Zeichenkette | Ja | Angabe einer Fehlerbeschreibung bei Übergabe der Statuscodes 400, 404 oder 500 |
| Status | Zeichenkette | Ja | Angabe des Statuscode (400, 404 oder 500) |

Beispieldatensatz für das setzen der Zahlsperre:

Nachfolgend ist ein Beispiel für die Übermittlung der Daten im beschriebenen Format dargestellt, um einen neuen Eigentümer an einer Abnahmestelle zu hinterlegen:

Content-Type: application/sag.zahlsperre.v1+json

```
{  
    "ZahlsperreBis": "2022-12-31",  
    "Grund": "Begründung weshalb die Zahlsperre gesetzt wurde"  
}
```

Beispieldatensatz für eine Rückmeldung bei fehlgeschlagenem Aufruf:

Der HTTP Status-Code ist im Fehlerfall immer im Bereich 400 – 500.

Content-Type: application/sag.fehlermeldung.v1+json

```
{  
    "Fehlertext": "Der angegebene Mandant '100' ist in der Datenquelle nicht vorhanden oder  
    Sie verfügen nicht über die nötigen Rechte. Es wurde kein Benutzerprofil angegeben. Kein dem  
    aktuellen Benutzer zugeordnetes Profil hat Zugriff auf den Mandanten.",  
    "Status": "404"  
}
```

2.3 Detailliertes Fehler- und Diagnoseprotokoll

Das Programm **Schnittstelle Prozessanbindung** besitzt eine flexible Schnittstelle zur Konfiguration von Protokoll- und Debug-Funktionalität. Die Einrichtung und Konfiguration der Protokollierung erfolgt in den Dateien *log4net.service.config* im Installationsverzeichnis der Schnittstelle.

Im Auslieferungszustand sind im Allgemeinen die folgenden Protokollziele konfiguriert:

- **WINDOWS EREIGNISANZEIGE**
Alle auftretenden kritischen Fehler werden in der Ereignisanzeige im Protokoll **Anwendung** mit dem Quellennamen **Schnittstelle Prozessanbindung** aufgezeichnet.
- **PROTOKOLLDATEI**
Das Programm legt eine Protokolldatei im Verzeichnis **Schnittstelle Prozessanbindung** mit dem Namen *Schnittstelle Prozessanbindung.service.log* an. In dieser Datei werden Fehler, Warnungen und Informationen abgelegt. Damit lässt sich der Programmablauf z.B. für die Fehlersuche nachvollziehen.
- **WINDOWS TRACE SCHNITTSTELLE**
Diese Ausgabe ist vorrangig zur Fehleranalyse eingerichtet. Alle Informationen werden in die Windows-eigene Trace-Schnittstelle geschrieben und können z.B. mit dem Tool DebugView von Microsoft ausgewertet werden.

Diese Liste kann beliebig vergrößert und verkleinert werden, je nach aktuellem Bedarf der Protokollierung. An dieser Stelle wird auf die Dokumentation der verwendeten Logging-Komponente log4net verwiesen.

Bitte beachten Sie, dass das Diagnoseprotokoll vorrangig zur Fehleranalyse durch den Kundenservice vorgesehen ist und es sich nicht um ein Anwenderprotokoll handelt. Sie können Anfragen an den Kundenservice zu Fehlersituationen unterstützen, indem Sie die erzeugten Fehlerprotokolldateien immer mit zur Verfügung stellen.

Anhang A. Support für Individualprojekte



Schleupen SE

Cottaer Straße 4

01159 Dresden



kundenservice@schleupen.de

Betreff: IVP Produkt Schnittstelle Prozessanbindung

Anhang B. Versionshistorie

Versionshistorie

| Version | Aktivität | | | Status: vorgelegt abgelehnt in Arbeit freigegeben |
|---------|------------|-------|---|---|
| | am | durch | Beschreibung | |
| 1.0.0.0 | 11.10.2022 | VKA | Erstellung | vorgelegt |
| 1.0.3.0 | 15.02.2023 | VKA | Aktualisierung | vorgelegt |
| 1.0.3.0 | 25.04.2023 | MJA | Ergänzung der Einrichtung des Vertragsdatenmanagers | vorgelegt |
| 1.0.4.0 | 24.05.2023 | MJA | Weitere Personenänderungen ergänzt | vorgelegt |