

Pflichtenheft:

Microservice zum Daten-Transfer von App-Betreibern an Kundenvertragspartner

Im Projektkontext auch als „Meldesystem“ oder „Middleware“ bezeichnet.

Version 1.6

Stand: 07.02.2023

Autoren:

Konstantin Schaible (SSB AG)

René Piesker (VVS GmbH)

Daniel Holz (bodo GmbH)

Werner Mies (DING GmbH)

Harald Bender (NVBW GmbH)

Tim Weiskopf (NVBW GmbH)

Sebastian Neifer (NVBW GmbH)

Andreas Schmid (Almato AG)

Dr. Tobias Henninger (NVBW GmbH)

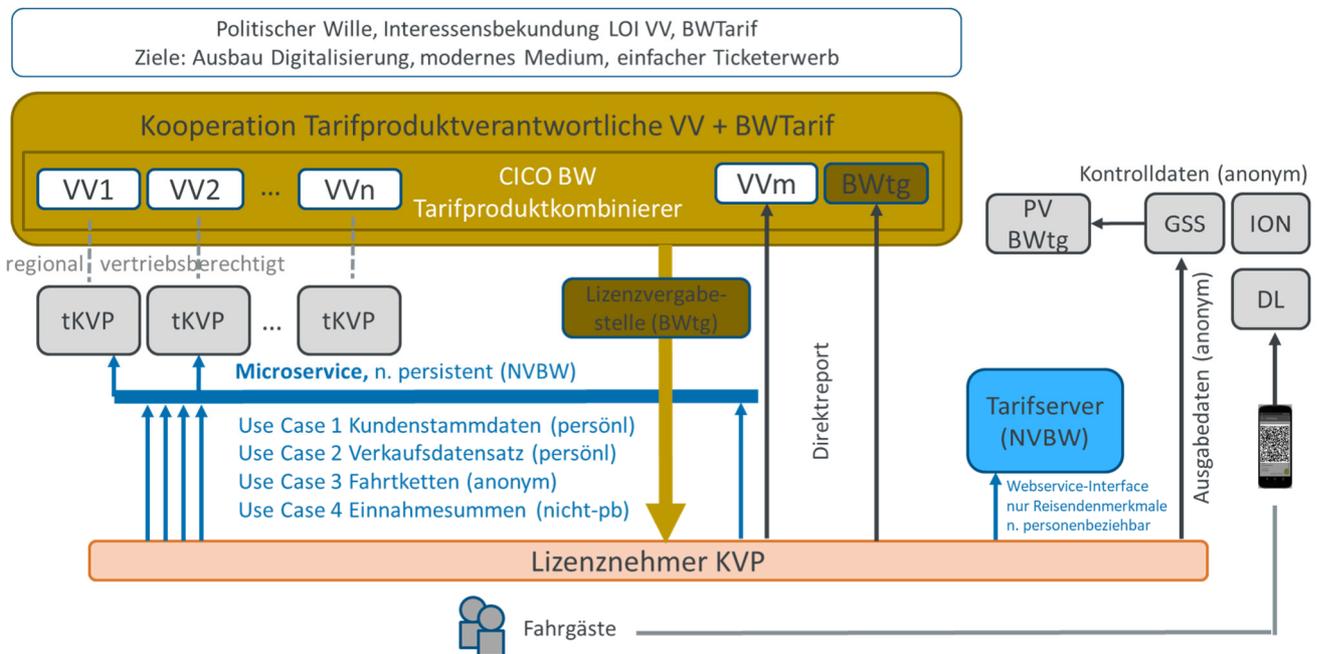
Version	Datum	Verantwortlicher	Anmerkungen
1.0	29.09.2021	S. Neifer	
1.1	31.07.2022	S. Neifer	
1.2	25.10.2022	A. Schmid	Korrektur der JSON-Beispieldaten
1.3	06.12.2022	T. Henninger	Übersichtsbild erneuert und Einführung zu den Uses Cases verfasst
1.4	07.12.2022	A. Schmid	Datenformat vereinheitlicht, Beispiele angepasst, Beschreibung Bool Attribute angepasst.
1.5	09.12.2022	T. Henninger	Testumgebung eingesetzt
1.6	07.02.2023	T. Henninger	Opt-In-Vorgaben eingesetzt

Inhalt

Einführung und Übersicht CICO BW	4
Use Cases.....	5
Übersicht	5
Datenschutz und Opt-Ins in der Endnutzung	6
Use Case 1: Kundenstammdaten	7
Häufigkeit der Übertragung	7
Daten	7
Use Case 2: Verkaufsdatensätze (Marketing/Vertrieb + Abrechnung/Einnahme)	9
Häufigkeit der Übertragung mindestens einmal im Monat kumuliert.....	10
Daten	10
Use Case 3: Fahrtketten-Datensätze:	13
Häufigkeit der Übertragung mindestens einmal im Monat kumuliert	13
Daten	13
Use Case 4: Einnahmesumme-Datensätze:.....	16
Häufigkeit der Übertragung mindestens einmal im Monat kumuliert.....	16
Daten	16
Technische Datenübermittlung.....	19
a) App-Betreiber (Lizenznehmer) an Microservice (NVBW).....	19
b) Microservice an territorialen Kundenvertragspartner (KVP)	19
Datenstruktur	19
Häufigkeit der Datenübertragung	20
c) Anforderung an Datenabnehmer/Ziel.....	20
Anforderung an Ziel der Daten.....	20
d) Fehlerhandling.....	20
Testumgebung SwaggerUI.....	21

Einführung und Übersicht CICO BW

Das Übersichtsbild zeigt die funktionale Gesamtarchitektur von CICO BW und die Verortung des hier beschriebenen Microservices.



Die Politik hat sich zum Ziel gesetzt, die Digitalisierung speziell auch im Mobilitätssektor voranzubringen und den Modal Split-Anteil des ÖPNV bis 2030 deutlich zu erhöhen (ÖPNV-Strategie 2030 des Ministeriums für Verkehr Baden-Württemberg). Dazu verhelfen einfache Kundenzugänge in Form moderner Medien wie Smartphones und ein Abbau von Zugangshemmnissen bei Ticketerwerb und in der Tariflandschaft.

Vor diesem Hintergrund hat sich eine Kooperation der Verkehrsverbünde (VV) und der Baden-Württemberg-Tarif GmbH (BWtg) gebildet, um im Land ein Check-In-Check-Out-Produkt auf den Markt bringen (CICO BW). Fahrgäste sollen bei Fahrtantritt auf ihrem Smartphone einchecken und am Fahrtende wieder auschecken, ohne sich Gedanken über die einzelnen Tarife machen zu müssen. Bei CICO BW handelt es sich nicht eigentlich um ein neues Tarifprodukt, sondern um einen Tarifproduktkombinierer. Dieser kombiniert einzelne Tarifprodukte der Verbünde und des BW Tarifs so, als hätten sich die Fahrgäste die Einzelprodukte in einer optimalen Weise selbst zusammengekauft. Die Logik dazu ist in einem Tarifserver untergebracht, der von Fraunhofer IVI geliefert und bei der NVBW gehostet wird. Verbünde und die BWtg sind dabei Produktverantwortliche (PV) im Sinne der VDV-KA – und zwar für ihre je eigenen zugelieferten Teilkomponenten. Die CICO-Kooperation hat entschieden, CICO BW nicht selbst zu vertreiben, sondern die Vertriebsrechte dazu an beliebig viele Lizenznehmer zu vergeben. Zur Unterstützung des Lizenzierungsprocedere hat die BW Tarif GmbH die Rolle einer Lizenzvergabe-stelle eingenommen. Als Fahrtberechtigung wird ein VDV-konformer 2D-Barcode auf dem Smartphone der Fahrgäste ausgegeben. Eine Kontrolle erfolgt mit der Infrastruktur, wie sie sich bereits für die elektronische Kontrolle des BW Tarifs im Aufbau befindet (rechter AST im Schaubild mit der VDV-KA-Rolle der kontrollierenden Dienstleister (DL), in der Regel die Verkehrsunternehmen. Sie speisen ihre Kontrolldaten über das Interoperabilitätsnetzwerk (ION) der VDV-KA bzw. die regionale Vermittlungsstelle (GSS) in die PV-Systeme der Produktverantwortlichen, sofern diese vorhanden sind.

Die Eigenschaft von CICO BW als ein Kombinierer von einzelnen Tarifprodukten bringt es nun mit sich, dass die vertreibenden Lizenznehmer ihre Einnahmen mit genau jedem einzelnen Tarifproduktgeber abrechnen müssen, dessen Produkte sie verkauft haben. Liegt die Reisekette innerhalb eines Verbundes, kommen die jeweiligen Verbundtarife zur Anwendung (z. B. VVS), ist die Reisekette verbundübergreifend, so gilt der BW Tarif. Lizenzgeber müssen somit gegenüber mehr als 20 einzelnen Tarifproduktgebern (Produktverantwortlichen PV) abrechnen und die entsprechenden Daten übermitteln. BW Tarif GmbH setzt dabei auf die direkte bilaterale Datenübermittlung, wie sie bereits aus dem Verkauf von BW Tarif besteht. Auch einzelne Verbünde setzen auf eine bilaterale Abrechnung, vor allem, wenn sie selbst für den Vertrieb verantwortlich sind. In der Regel jedoch übernehmen in einem Verbund andere Organisationen den Vertrieb: Die für sie handelnden territorialen Kundenvertragspartner (tKVP).

Damit CICO-Lizenznehmer nun nicht die Daten sämtlicher tKVP an jeden einzelnen tKVP separat verschicken müssen, wird dieser Microservice eingerichtet. Er übernimmt die Aufgabe des Versands der Daten an die richtigen Empfänger. Als Unterscheidungsmerkmal, an welchen tKVP die Daten versandt werden sollen, bedient er sich der ORG_ID des PV, also des dahinter stehenden tarifgebenden Verkehrsverbunds. Die ORD_ID ist ein Datum des eTicketing-Standards der VDV-Kernapplikation.

Use Cases

Übersicht

Die Lizenznehmer haben an die tKVP Daten zur Abrechnung, zum Support und zu Marketingzwecken zu übermitteln. Dazu wurden im Einzelnen die folgenden vier Use Cases definiert:

1. Use Case 1: Kundenstammdaten (persönlich)
2. Use Case 2: Verkaufsdatensätze (personenbeziehbar)
3. Use Case 3: Fahrtketten (anonym)
4. Use Case 4: Einnahmesummen (nicht personenbeziehbar)

Zur Motivation der einzelnen Use Cases kann folgendes ausgesagt werden: Zur monatlichen Abrechnung in jedem Falle benötigt werden die Monatssummen in Use Case 4. Die tKVP werden diese in die verbundinterne Einnahmeaufteilung einspeisen müssen. Da viele tKVP die Einnahmen jedoch aus Einzelverkäufen aufsummieren und nicht zuletzt zu Marketingzwecken Kenntnis ihrer Kunden haben möchten, wird Use Case 2 benötigt, der den Verkaufsdatensatz mit den Einzelverkäufen enthält. Die zugehörigen und nicht häufig wechselnden Kundenstammdaten sind in Use Case 1 gefasst. Die anonymen Fahrtketten von Use Case 3 werden zu Zwecken der Verkehrs- und Produktplanung erfasst.

Wichtig: Es darf aus Datenschutzgründen keine Speicherung von Daten im Microservice der NVBW erfolgen. Die Datenverarbeitung muss nicht-persistent vonstattengehen.

Im Folgenden soll nun der Inhalt und die Struktur der einzelnen Use Case beschrieben werden. Werden persönliche Daten via Microservice übermittelt, so ist stets auch die Kundeneinwilligung zu übermitteln. Genauere Betrachtungen zu den Opt-Ins bzw- Opt-Outs werden daher im Folgenden noch vorangestellt.

Datenschutz und Opt-Ins in der Endnutzung

Es besteht bei den lizenzgebenden Verbänden (Produktverantwortliche PV) der Wunsch, Kundendaten aus Fahrscheinverkäufen für Marketing und individualisierte Werbekampagnen einzusetzen. Sie begründen dies datenschutzrechtlich mit dem Zweck eines „berechtigten Interesses“. (Lediglich die ebenfalls als PV fungierende BWTG signalisierte kein Interesse an einer diesbezüglichen Verwendung von Kundendaten.) Der Landesbeauftragten für Datenschutz und Informationsfreiheit (LfDI) konnte der Argumentation eines „berechtigten Interesses“ jedoch nicht folgen und hat eine Empfehlung ausgesprochen, die Weitergabe von Kundendaten von den Lizenznehmern zu den tKVP an Opt-Ins der Fahrgäste zu knüpfen. Die Lizenzgeber haben sich dieser Empfehlung angeschlossen.

Der Microservice ist daher von den Lizenznehmern zur Weitergabe so einzusetzen, dass persönliche Daten nur bei vorliegenden Opt-Ins an die tKVP weitergegeben werden. Für die Aufnahme der verschiedenen Kundeneinwilligungen wurden in Use Case 1 die üblichen Felder in die Übergabestruktur aufgenommen. Dies sind:

- Einwilligung zur telefonischen Kontaktaufnahme (Opt-In)
- Einwilligung zur Kontaktaufnahme per Email (Opt-In)
- Einwilligung der Datennutzung zur Marktforschung (Opt-In)
- Verweigerung postalischer Werbung (Opt-Out)

Der Use Case 1 Kundenstammdaten muss daher nur übermittelt werden, wenn mind. eine der oben genannten Einwilligungen oder keine Verweigerung der postalischen Werbung vorliegt. Zusätzlich darf selbstverständlich kein kundenseitiger Wunsch zur Datenlöschung vorliegen. Wird Use Case 1 übermittelt, gelten die folgenden Einschränkungen:

- Übermittlung der Telefonnummer nur bei entsprechendem Opt-In
- Übermittlung der Email-Adresse nur bei entsprechendem Opt-In
- Übermittlung der Wohnadresse nur ohne Verweigerung postalischer Werbung

Use Case 2 übermittelt den Verkaufsdatensatz granular je Verkaufsvorgang. Bis auf die KundenID bei den Verkäufen enthält Use Case 2 keine persönlichen Daten. Der Verkaufsdatensatz ist allerdings personenbeziehbar, sofern die KundenID übermittelt wird. Er wird damit zu seinem persönlichen Verkaufsdatensatz. Ein persönlicher Verkaufsdatensatz wird jedoch nur zum Zwecke von Marketing und Vertrieb übermittelt. Daher ist ein Opt-In bei Marktforschung die zwingende Voraussetzung, dass bei Use Case 2 die KundenID übermittelt werden darf.

Doch selbst wenn die KundenID in Use Case 2 nicht übermittelt wird, muss Use Case 2 ohne KundenID übermittelt werden, denn ein tKVP benötigt diese Daten für die Abrechnung gegenüber dem PV, in der Regel zur Einspeisung in die Einnahmeaufteilung (Ausnahme: ein tKVP begnügt sich mit den Einnahmesumme in Use Case 4). Daher muss Use Case 2 alle abrechnungsrelevanten Daten eines Verkaufsdatensatzes enthalten (Verkaufszeitpunkt, Relation, Eindeutigkeit des Fahrscheins). Diese Information ist auch in den anonymen Daten von Use Case 2 ohne KundenID enthalten.

Use Case 1: Kundenstammdaten

Übermittlung der Kundenstammdaten vom Lizenznehmer zum territorial verantwortlichen Kundenvertragspartner.

Beispiel:

- Lizenznehmer betreibt App,
- Kunde registriert sich in App, fährt im Verbundgebiet der VVS GmbH,
- Kundenstammdaten werden über den Microservice der NVBW an die SSB AG übertragen.

Häufigkeit der Übertragung
mindestens 1x monatlich

Daten

Name	Key	Type	Länge	Value	Beschreibung
Header	Pflicht	String	30	Kunde	Legt fest, welcher Nachrichtenart gesendet wird
KvpOrgID	Pflicht	Integer	5		Org-ID nach VDV-KA des Lizenznehmers
PvOrgID	Pflicht	Integer	5		Org-ID des Verkehrsverbundes in dem das Ticket verkauft wird
AbrechnungsPeriode	Pflicht	Date (mm-yyyy)	7	08-2021	Monat für welchen die Tickets abgerechnet wurden, z.B. 08-2021
Kundenliste					JSON Struktur der Kunden

Name	Key	Type	Länge	Value	Beschreibung
KundenID	Pflicht	String	50		Eindeutige sich nicht verändernde Kundennummer des
					Benutzers beim Lizenznehmer
Titel	Optional	String	30		Titel der Person
Name	Pflicht	String	50		Nachname der Person
Vorname	Pflicht	String	50		Vorname der Person
Strasse	Optional	String	50		Adresse Straßename
Hausnummer	Optional	String	30		Adresse Hausnummer

Postleitzahl	Optional	String	30		Adresse Postleitzahl
Ort	Optional	String	30		Adresse Wohnort
Land	Optional	String	30		Länderklartext wenn abweichend von Deutschland. Bei Deutschland nicht notwendig.
TelNr	Optional	String	50		Telefonische Erreichbarkeit der Person
Mailadresse	Optional	String	50		Erreichbarkeit per Email der Person
Geburtsdatum	Pflicht	Date	10		Geburtsdatum im Format dd.mm.yyyy
EinwilligungWerbungTel	Pflicht	Boolean	1		Einwilligung zur telefonischen Kontaktaufnahme TRUE oder FALSE
EinwilligungWerbungEmail	Pflicht	Boolean	1		Einwilligung zur Kontaktaufnahme per Email
EinwilligungMarktforschung	Pflicht	Boolean	1		Einwilligung zu Marktforschungszwecken
EntzugWerbungPost	Pflicht	Boolean	1		Verweigerung postalischer Werbung
Aenderungszeitpunkt	Pflicht	Datetime (dd.mm.yyyy hh:MM:ss)	19		Datum und Uhrzeit, zu welcher der Benutzer zuletzt geändert wurde
Loeschkennzeichen	Pflicht	Boolean	1		Information ob der Kundendatensatz gelöscht werden soll TRUE oder FALSE

Beispiel:

```
curl --location --request POST 'https://0aaf1vt0c6.execute-api.eu-central-1.amazonaws.com/test/customer' \
--header 'Content-Type: application/json' \
--header 'x-api-key: WZaXTzjOju3OCIH3ZqL0DaRUrZ6Qd9F3a2UsStUw' \
--data-raw '{
  "Header": "Kunde",
  "KvpOrgID": 6233,
  "PvOrgID": 6234,
  "Abrechnungsperiode": "08-2021",
  "Kundenliste": [
    {
```

```

    "KundenID": "Test123",
    "Titel": "Dr.",
    "Name": "Mustermann",
    "Vorname": "Max",
    "Strasse": "Musterstraße",
    "Hausnummer": "37/F",
    "Postleitzahl": "54321",
    "Ort": "Musterstadt",
    "Land": "Schweiz",
    "TelNr": "+41 5461 6441258",
    "Mailadresse": "test@mail.ch",
    "Geburtsdatum": "01.03.2001",
    "EinwilligungWerbungTel": true,
    "EinwilligungWerbungEmail": false,
    "EinwilligungMarktforschung": false,
    "EntzugWerbungPost": false,
    "Aenderungszeitpunkt": "27.08.2021 13:24:45",
    "Loeschkennzeichen": false
  },
  {
    "KundenID": "Test124",
    "Titel": "Dr. Prof. ",
    "Name": "Musterfrau",
    "Vorname": "Ruth",
    "Strasse": "Musterstraße",
    "Hausnummer": "31/F",
    "Postleitzahl": "34567",
    "Ort": "Musterstadt",
    "TelNr": "+41 5461 6441258",
    "Mailadresse": "test@mail.de",
    "Geburtsdatum": "01.03.2001",
    "EinwilligungWerbungTel": false,
    "EinwilligungWerbungEmail": false,
    "EinwilligungMarktforschung": false,
    "EntzugWerbungPost": true,
    "Aenderungszeitpunkt": "27.08.2021 13:24:45",
    "Loeschkennzeichen": false
  }
]
}
'

```

Use Case 2: Verkaufsdatensätze (Marketing/Vertrieb + Abrechnung/Einnahme)

Übermittlung der einzelnen Verkaufstransaktionen vom Lizenznehmer zum territorial verantwortlichen Kundenvertragspartner, für die Ermittlung der Werte zur Meldung an die Einnahmeverteilung (Einnahmemeldung) und zu Zwecken des Marketing.

Beispiel:

- Lizenznehmer betreibt App,
- Kunde fährt mit App im Verbundgebiet der VVS GmbH,
- Gefahrene Fahrtkettendaten führen zu einem VVS Tarifprodukt

- Ein Ticket der VVS GmbH wird dem Kunden abgerechnet
- Abrechnungsdaten werden über die NVBW an die SSB AG übertragen. Verkaufte Tarifprodukte werden über die NVBW an die SSB AG übertragen.
- Die SSB nutzt die Übermittlung der verkauften Tarifprodukte für Einnahmemeldungen
- Die SSB kann die Verkaufstransaktionsdaten für Marketing und Vertrieb nutzen

Häufigkeit der Übertragung
mindestens einmal im Monat kumuliert

Daten

Kopf

Name "Feld im Tarifserver"	Key	Type	Länge	Value	Beschreibung
Header	Pflicht	String	30	Verkauf	Legt fest, welcher Nachrichtenart gesendet wird
KvpOrgID	Pflicht	Integer	5		Org-ID nach VDV-KA des Lizenznehmers
PvOrgID	Pflicht	Integer	5		Org-ID des Verkehrsverbundes
AbrechnungsID	Pflicht	String	30		Abrechnungskopffidentifikation (eindeutige ID der Abrechnung in der Abrechnungsperiode)
AbrechnungsPeriode	Pflicht	Date (mm-yyyy)	7	08-2021	Monat für welchen die Tickets abgerechnet wurden, z.B. 08-2021
Verkaufsliste					JSON Struktur mit Verkäufen

Verkaufe

Name "Feld im Tarifserver"	Key	Type	Länge	Value	Beschreibung
AbrechnungsPositionID	Pflicht	Integer	20		Identifiziert den Datensatz
KundenID	Optional	String	50		Kundennummer des Benutzers beim Lizenznehmer (bei Werbesperre Marktforschung nicht übermitteln)
TicketTypeID	Pflicht	String	20		ID des Tickettyps, z.B. 101 oder Fehlercode für unbekannte Restposten (z.B. aus Tarifserverfehlerdifferenzen)

TicketTypDesc	Pflicht	String	50		Bezeichnung des Tickettyps, z.B.: Einzelticket Erwachsener oder Erstattung
PreisStufe	Optional	String	20	3	Preisstufe des Produktes
PreisEinzel	Pflicht	Integer	12	540	Preis für ein Ticket in Cent ohne Trennzeichen
StartAbrechnung	Optional	String	50	100 (Ulm)	Start des abgerechneten Tarifprodukts
ZielAbrechnung	Optional	String	50	300 (Erbach)	Ziel des abgerechneten Tarifprodukts
RaumlicheGueltigkeit	Optional	String	75		Räumliche Gültigkeit des Tarifproduktes
BeginnGueltigkeit	Pflicht	Datetime (dd.mm.yyyy hh:MM:ss)	19	10.08.2021 12:23:42	Gültigkeitsbeginn des Tickets (Date-Time) oder Erstattungszeitpunkt in local time
EndeGueltigkeit	Pflicht	Datetime (dd.mm.yyyy hh:MM:ss)	19	10.08.2021 12:23:42	Gültigkeitsende des Tickets (Date-Time) oder Erstattungszeitpunkt in local time
Verkaufszeitpunkt	Pflicht	Datetime (dd.mm.yyyy hh:MM:ss)	19	10.08.2021 12:23:42	Zeitpunkt des Verkaufs z.B. 3Uhr Nachts (Bestpreisoptimierung) oder Erstattungszeitpunkt in local time
TarifVersion	Pflicht	String	50	PVTM_6234_G200401_0000_E7_1_1.6_VVS	Aktuelle Version des ProduktModuls des PV
VertriebsKanal	Optional	String	50	Fairtiq grün (oder Tickin, oder wasweissich)	Wird durch Whitelabelvergabe definiert, z.B. bwegt App, DB-App, VVSApp, etc.
Storno	Optional	String	1		S=Storno, F=Fehlbon(Maschinen storno)

					identifiziert, ob und wenn ja um welche Art von Storno es sich handelt
StornoGrund	Pflicht wenn Storno gefüllt	String	30	CheckOut nicht korrekt erfasst	Weshalb wird storniert
Gutschrift	Optional	String	1	T	T = Tarifserverfehler Ggf. weitere
Erstattungsvorgangskennung	optional	String	30	Y143YZ Bad Canstatt - Ulm	Der vom PV benannte Fehlercode und Hinweistext
Testdatensatz	Optional	Boolean	1		Die Möglichkeit die Übermittlung und Verarbeitung zu testen. Der Datensatz wird dann in der Abrechnung nicht berücksichtigt

Beispiel:

```
curl --location --request POST 'https://0aaflvt0c6.execute-api.eu-central-1.amazonaws.com/test/sale' \
--header 'Content-Type: application/json' \
--header 'x-api-key: WZaXTzjOju3OCIH3ZqL0DaRUrZ6Qd9F3a2UsStUw' \
--data-raw '{
  "Header": "Verkauf",
  "KvpOrgID": 6233,
  "PvOrgID": 6234,
  "AbrechnungsID": "Abrechnungskopfidentifikation",
  "AbrechnungsPeriode": "08-2021",
  "Verkaufsliste": [
    {
      "AbrechnungsPositionID": 1,
      "KundenID": "123Test",
      "TicketTypID": "050301",
      "TicketTypDesc": "EinzelTagesTicket",
      "PreisStufe": "1",
      "PreisEinzel": 532,
      "StartAbrechnung": "100(Ulm)",
      "ZielAbrechnung": "300(Erbach)",
      "RaeumlicheGueltigkeit": "3-4-5",
      "BeginnGueltigkeit": "10.08.2021 12:23:42",
      "EndeGueltigkeit": "11.08.2021 07:00:00",
      "Verkaufszeitpunkt": "11.08.2021 03:00:00",
      "TarifVersion": "PVTM_6234_G200401_0000_E7_1_1.6_VVS",
      "VertriebsKanal": "1",
      "Storno": "S",
      "StornoGrund": "CheckOut nicht korrekt erfasst",

```

```

    "Gutschrift": "T",
    "Erstattungsvorgangskennung": "Y143YZ Bad Canstatt - Ulm",
    "Testdatensatz": true
  },
  {
    "AbrechnungsPositionID": 2,
    "KundenID": "123ULM",
    "TicketTypID": "041201",
    "TicketTypDesc": "EinzelTicket",
    "PreisStufe": "1",
    "PreisEinzel": 632,
    "StartAbrechnung": "100 (Ulm) ",
    "ZielAbrechnung": "300 (Erbach) ",
    "RaeumlicheGueltigkeit": "100-200-300",
    "BeginnGueltigkeit": "16.08.2021 12:23:42",
    "EndeGueltigkeit": "16.08.2021 14:23:42",
    "Verkaufszeitpunkt": "17.08.2021 03:00:00",
    "TarifVersion": "PVTM_6234_G200401_0000_E7_1_1.6_VVS",
    "VertriebsKanal": "1"
  }
]
}
'

```

Use Case 3: Fahrtketten-Datensätze:

Übermittlung der einzelnen Fahrtketten von Lizenznehmer zum territorial verantwortlichen Kundenvertragspartner. Nur für die Verkaufsdaten, nicht für Stornos.

Beispiel:

- Lizenznehmer betreibt App,
- Kunde fährt mit App im Verbundgebiet der VVS GmbH,
- App ermittelt die gefahrenen Fahrtketten
- Gefahrene und abgerechnete Fahrtkettendaten werden über die NVBW an die SSB AG übertragen, kein Kundendatenbezug.

Häufigkeit der Übertragung

mindestens einmal im Monat kumuliert

Daten

Kopf

Name "Feld im Tarifserver"	Key	Type	Länge	Value	Beschreibung
Header	Pflicht	String	30	Fahrtketten	Legt fest, welcher Nachrichtenart gesendet wird
KvpOrgID	Pflicht	Integer	5		Org-ID nach VDV-KA des Lizenznehmers

PvOrgID	Pflicht	Integer	5		Org-ID des Verkehrsverbundes
AbrechnungsPeriode	Pflicht	Date (mm-yyYY)	7	08-2021	Monat für welchen das Ticket abgerechnet wurde, z.B. 08-2021
Ticketliste					JSON Struktur mit Tickets

Tickets

Name "Feld im Tarifserver"	Key	Type	Länge	Value	Beschreibung
TicketTypeID	Pflicht	String	20		ID des Tickettyps, z.B. 101
TicketTypDesc	Pflicht	String	50		Bezeichnung des Tickettyps, z.B.: Einzelticket Erwachsener
PreisStufe	Optional	String	20	3	Preisstufe des Produktes
PreisEinzel	Pflicht	Integer	12	540	Preis für das Ticket in Euro-Cent ohne Trennzeichen
Fahrtenliste					JSON Struktur mit Fahrten

Fahrten

Name "Feld im Tarifserver"	Key	Type	Länge	Value	Beschreibung
LinieNr	Pflicht	String	50		Welche Linie wurde verwendet
VerkehrsmittelName	Pflicht	String	50		NVBW-Verkehrsmittelname
VerkehrsmittelNr"	Pflicht	Integer	2		NVBW-Verkehrsmittelnummer
Haltestellenliste					JSON Struktur mit Haltestellen

Haltestellen

Name "Feld im Tarifserver"	Key	Type	Länge	Value	Beschreibung
Ankunft	Optional	DateTime (dd.mm.yyy hh:MM:ss)	19		Uhrzeit der Ankunft an einer Haltestelle in local time

Abfahrt	Optional	DateTime (dd.mm.yyyy hh:MM:ss)	19		Uhrzeit der Abfahrt an einer Haltestelle in local time
Haltestelle	Pflicht	String	50		DHID GlobalID
Haltestellenname	Pflicht	String	50		Name der Haltestelle

Beispiel:

```
curl --location --request POST 'https://0aaflvt0c6.execute-api.eu-central-1.amazonaws.com/test/drivechain' \
--header 'Content-Type: application/json' \
--header 'x-api-key: WZaXTzjOju30CIH3ZqL0DaRUrZ6Qd9F3a2UsStUw' \
--data-raw '{
  "Header": "Fahrketten",
  "KvpOrgID": 6235,
  "PvOrgID": 6234,
  "AbrechnungsPeriode": "08-2021",
  "Ticketliste": [
    {
      "TicketTypID": "20151",
      "TicketTypDesc": "EinzelTicket Erwachsene",
      "PreisStufe": "Zone 32",
      "PreisEinzel": 255,
      "Fahrtenliste": [
        {
          "LinieNr": "24 Ostendplatz",
          "VerkehrsmittelNr": 42,
          "VerkehrsmittelName": "Bus",
          "Haltestellenliste": [
            {
              "Ankunft": "16.07.2022 06:14:50",
              "Abfahrt": "16.07.2022 06:14:50",
              "haltestelle": "de:08111:6332",
              "HaltestellenName": "Bad Cannstatt Wilhelmsplatz"
            }
          ]
        }
      ]
    }
  ]
}
```

Use Case 4: Einnahmesumme-Datensätze:

Übermittlung der Summe der Einnahmen je verkaufte Ticketart vom Lizenznehmer zum territorial verantwortlichen Kundenvertragspartner.

Ziel: Abrechnungsbasis zwischen Lizenznehmer und KVP/PV

Häufigkeit der Übertragung

mindestens einmal im Monat kumuliert

Daten

Kopf

Name	Key	Type	Länge	Value	Beschreibung
Header	Pflicht	String	30	Einnahmen	Legt fest, welcher Nachrichtenart gesendet wird
KvpOrgID	Pflicht	Integer	5		Org-ID nach VDV-KA des Lizenznehmers
PvOrgID	Pflicht	Integer	5		Org-ID des Verkehrsverbundes
AbrechnungsDatumAb	Pflicht	Date (dd.mm.yyyy)	10	01.08.2021	Erster Tag der Abrechnungsperiode
AbrechnungsDatumBis	Pflicht	Date (dd.mm.yyyy)	10	10.08.2021	Letzter Tag der Abrechnungsperiode
AbrechnungsID	Pflicht	String	20	Af15054g	Eindeutige ID der Abrechnung
AbrechnungsPeriode	Optional	Date (mmyyyy)	7	08-2021	Monat für welchen das Ticket abgerechnet wurde, z.B. 08-2021
SummeAllerVerkäufe	Pflicht	Integer	12		Summierter Preis der Menge aller Tickets aus den Einnahmen in Cent ohne Komma, kann mit Stornos negativ werden
Einnahmenliste					JSON Struktur der Einnahmen

Einnahmen

TicketTypID	Pflicht	String	20		Zeilenidentifizier ist TicketID, z.B. „101“ oder Fehlercode für unbekannte Restposten (z.B. aus
-------------	---------	--------	----	--	---

					Tarifserverfehlerdifferenzen)
TicketTypDesc	Pflicht	String	50		Bezeichnung des Tickettyps, z.B.: Einzelticket Erwachsener
PreisStufe	Optional	String	20	3	Anzahl der Zonen
PreisEinzel	Pflicht	Integer	12	540	Preis für ein Ticket in Cent ohne Komma, immer positiver Wert
TarifVersion	Pflicht	String	50	PVTM_6234_G200401_0000_E7_1_1.6_VVS	Aktuelle Version des ProduktModuls des PV
Menge	Pflicht	Integer	12		Menge der Tickets, kann bei Stornos negativ werden
PreisGesamt	Pflicht	Integer	12		Summierter Preis der Menge der Tickets in Cent ohne Komma, kann bei Stornos negativ werden

Beispiel:

```
curl --location --request POST 'https://0aaf1vt0c6.execute-api.eu-central-1.amazonaws.com/test/revenue' \
--header 'Content-Type: application/json' \
--header 'x-api-key: WZaXTzjOju3OCIH3ZqL0DaRUrZ6Qd9F3a2UsStUw' \
--data-raw '{
  "Header": "Einnahmen",
  "KvpOrgID": 6235,
  "PvOrgID": 6234,
  "Tarifversion": "PVTM_6234_G200401_0000_E7_1_1.6_VVS",
  "AbrechnungsDatumAb": "01.08.2021",
  "AbrechnungsDatumBis": "10.08.2021",
  "AbrechnungsID": "10-08-2ss021",
  "AbrechnungsPeriode": "08-2022",
  "SummeAllerVerkaeufe": 164376,
  "Einnahmenliste": [
    {
      "TicketTypID": "50254",
      "TicketTypDesc": "EinzelTagesTicket Online",
      "PreisStufe": "4",
      "PreisEinzel": 1044,
      "TarifVersion": "PVTM_6234_G200401_0000_E7_1_1.6_VVS",
      "Menge": 103,
      "PreisGesamt": 107532
    },
    {
      "TicketTypID": "50808",
      "TicketTypDesc": "StadtTicket EinzelTagesTicket",
      "PreisStufe": "108",
```

```
"PreisEinzel": 300,  
"TarifVersion": "PVTM_6234_G200401_0000_E7_1_1.6_VVS",  
"Menge": 99,  
"PreisGesamt": 29700  
},  
{  
  "TicketTypID": "20154",  
  "TicketTypDesc": "EinzelTicket Mobil",  
  "PreisStufe": "4",  
  "PreisEinzel": 522,  
  "TarifVersion": "PVTM_6234_G200401_0000_E7_1_1.6_VVS",  
  "Menge": 52,  
  "PreisGesamt": 27144  
}  
]  
}  
,
```

Technische Datenübermittlung

a) App-Betreiber (Lizenznehmer) an Microservice (NVBW)

Datenformat: JSON

Pro PvOrgID im Header muss gebündelt ein separater Datensatz gesendet werden. Z.B. Alle Kundenstammdaten der Periode. Keinesfalls soll jeder Kundendatensatz separat versendet werden.

Endpoint Testsystem:

Endpoint-URL	https://0aaf1vt0c6.execute-api.eu-central-1.amazonaws.com/test
API-Schlüssel	WZaXTzjOju3OCIH3ZqL0DaRUrZ6Qd9F3a2UsStUw
Content-Type	application/json

Die Endpoint URL zur Ansprache des Microservices bezieht sich auf das Testsystem bei Amazon Web Services. Es ist geplant, dass das Produktivsystem auf einen anderen Host umzieht. Die Zugangsdaten zum Produktivsystem werden nach erfolgreichem Test bereitgestellt.

b) Microservice an territorialen Kundenvertragspartner (KVP)

Konfiguration der Ziele

Konfiguration der Ziele wird mit einer yaml-Datei beschreiben. Der Betreiber des Microservice (NVBW) liefert hiervon eine Beispieldatei. Die tKVP liefern sodann ein auf ihre jeweiligen Wünsche angepasste Beispieldatei zurück, oder: Die tKVP äußern ihre Wünsche und übermitteln die URL und die Zugangsdaten (Benutzername und Zertifikat).

Ausgangsformate

Während für das Eingangsformat des Microservices stets JSON gefordert ist, lässt das Ausgangsformat folgende drei Stammformate zu:

- XML
- JSON
- CSV

Auf Basis dieser drei Stammformate lassen sich dann je Ziel-PV/tKVP das vom Ziel gewünschte Format ebenfalls in der yaml-Datei konfigurieren.

Datenstruktur

Datensätze werden vom Microservice pro PvOrgID getrennt ausgegeben.

Pro eingehender Datensatz, wird ein Datensatz ausgegeben. Die Datenstruktur der Eingangsfiles wird im Ausgang nicht zusammengeführt. Das Zielformat und die Reihenfolge der Daten, Feldnamen und die Anzahl der Felder entspricht in der Regel der Eingangsstruktur.

Erfolgt eine Ausgabe im cvs-Format, so lässt sich das gewünschte Zielformat beeinflussen. Die Reihenfolge, Feldnamen oder Anzahl der Felder kann pro PvOrgID separat festgelegt werden. Sollte die Reihenfolge, Feldnamen oder Anzahl nach Go-Live geändert werden müssen, ist eine Frist von 3 Monaten zur Produktivsetzung der benötigten Änderung einzuhalten. Produktivsetzungstermin wäre jeweils zum Quartalsende.

Häufigkeit der Datenübertragung

Die Häufigkeit der Datenübertragung richtet sich nach der Häufigkeit der Eingangsnachrichten.

c) Anforderung an Datenabnehmer/Ziel

Anforderung an Ziel der Daten

Die nachstehenden Anforderungen liegen im Verantwortungsbereich der jeweiligen tKVP auf Datenabnehmerseite:

Ausgabe auf gesichertes Ziellaufwerk des tKVP/PV (SFTP). Der Abnehmer muss dazu einen SFTP-Server bereitstellen. Weiter wird zum Auslesen der Daten ein SFTP-Client vom Datenabnehmer benötigt. Der Abnehmer ist für Datensicherheit und Datenschutz auf dem Ziellaufwerk verantwortlich (Sicherung mit Benutzernamen und Zertifikat z. B.). Die Dateibezeichnungsstruktur wird vom Ziel-KVP/PV vorgegeben (z.B. CiCo_“HeaderTyp“_“PV“_“KVP“_“Erstellungsdatum“.csv).

d) Fehlerhandling

Info an Lizenznehmer:

- Empfang am Microservice gescheitert
- Verarbeitung am Microservice gescheitert
 - o Validitätsprüfung mit Rückgabe des ersten Feldes, welches fehlerhaft war
 - Formatfehler (Datentyp)
 - Feldlängenprüfung
 - Pflichtfelder
 - o PvOrgID nicht gefunden
 - o Header Parameter falsch
- Zustellung im Zielsystem gescheitert

Lizenznehmer muss eine entsprechende Email-Adresse angeben, an welche die Info verschickt werden muss. Info an KVP/PV:

- Ist das Ziellaufwerk nicht erreichbar, muss eine sprechende Fehlermeldung an den KVP/PV per Email geschickt werden
- KVP/PV muss eine entsprechende Email-Adresse angeben, an welche die Info verschickt werden muss.

Was muss KVP/PV bzw. Lizenznehmer im Fehlerfall tun? Das Melde-System informiert nur, greift aber nicht korrigierend ein. Die Verantwortung zur Fehlerbehebung liegt auf Seiten der Lizenznehmer bzw. KVP/PV.

Ebenso notwendig ist die Bestätigung bei erfolgreicher Zustellung, automatisiert verarbeitbar durch Lizenznehmersystem.

Testumgebung SwaggerUI

Zu Entwicklungs- und Testzwecken des Microservice wurde der Datentransfer-Microservices auf dem Standard OpenAPI 3.0 dokumentiert. Basis hierfür ist eine sogenannte yaml-Datei. Diese beinhaltet neben den Aufrufen und Datenstrukturen nun auch die Dokumentation aus dem Pflichtenheft. Diese yaml-Datei ermöglicht den CICO BW-Lizenznehmer und den abnehmenden tKVP eine Einsicht in die API und eine Generierung der Client-Anwendungen. Die Datei wird den Lizenznehmern und tKVP zur Verfügung gestellt. Diese können dann die zur Anbindung notwendigen Clients generieren und in ihre Anwendung einbauen.

Darüber hinaus kann diese Beschreibungsdatei genutzt werden, die API zu visualisieren. Mit Hilfe von Swagger ist es einfach möglich die Dokumentation der API einzusehen und die einzelnen Aufrufe zu testen. Neben den einzelnen Services sind die Strukturen mit Beschreibung, Format- und Pflicht-Angaben aufgeführt.

Einsicht in die OpenAPI yaml-Datei unter folgendem Link: <https://nvbw-test.almato.com/openAPI.yaml>

URL der Swagger UI: <https://nvbw-test.almato.com/swaggerUI/>

Die Website ist gesichert mit einem Zertifikat. Die APIs mit entsprechenden API-Keys. Beide werden auf Anforderung zur Verfügung gestellt.

Startseite der Testumgebung:

