

**IBIS-IP Beschreibung der Dienste /
Service description**

**DIENST-TicketValidationService /
SERVICE-TicketValidationService
V2.3**

Gesamtbearbeitung

Ausschuss für Telematik und Informationssysteme (ATI)

Gefördert durch:



Bundesministerium
für Wirtschaft
und Energie

aufgrund eines Beschlusses
des Deutschen Bundestages

Das dieser VDV-Schrift zugrundeliegende Vorhaben IP-KOM-ÖV wurde mit Mitteln des Bundesministeriums für Wirtschaft und Energie unter dem Förderkennzeichen 19P10003 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

IBIS-IP Beschreibung der Dienste / Service description

DIENST-TicketValidationService /
SERVICE-TicketValidationService
V2.3

Sachbearbeitung

Unterausschuss für Telematik
(UA Telematik)

Autorenverzeichnis

René Fischli, Trapeze, Neuhausen
T. Kling, Trapeze, Neuhausen
Horst Sander, ATRON, Markt Schwaben
Tobias Huber, ATRON, Markt Schwaben

Der Anwender ist für die sorgfältige und ordnungsgemäße Anwendung der Schrift verantwortlich. Stellt der Anwender Gefährdungen oder Unregelmäßigkeiten im Zusammenhang mit der Anwendung dieser Schrift fest, wird eine unmittelbare Benachrichtigung an den VDV erbeten. Eine Haftung des VDV oder der Mitwirkenden an der Schrift ist, soweit gesetzlich zulässig, ausgeschlossen.

© Verband Deutscher Verkehrsunternehmen e. V. Köln 2015 | Alle Rechte, einschließlich des Nachdrucks von Auszügen, der fotomechanischen oder datenverarbeitungstechnischen Wiedergabe und der Übersetzung, vorbehalten.

Vorwort

Im Forschungsprojekt „Internet Protokoll basierte Kommunikationsdienste im ÖV - IP-KOM-ÖV“, gefördert vom Bundesministerium für Wirtschaft und Energie BMWi, wurde das Grundkonzept von IBIS-IP für die Fahrgastinformation entwickelt.

Das vorliegende Dokument spezifiziert den IBIS-IP-Dienst für die Ansteuerung eines Papier-Fahrschein-Entwerfers.

Die aktuelle Version V2.3 korrigiert die Beschreibung der Datentabellen in Kapitel 3.1.2, so dass sie den korrekten Datendefinitionen in der entsprechenden XSD V2.2 Datei entsprechen.

Foreword

In the research project "Internet Protocol based communication services in public transport - IP-KOM-ÖV", funded by the Federal Ministry of Economics and Energy BMWi, the basic concept of IBIS-IP for passenger information was developed.

This document specifies the IBIS IP service for controlling a paper ticket validator.

The current version V2.3 corrects the description of the data tables in chapter 3.1.2 so that they correspond to the correct data definitions in the corresponding XSD V2.2 file.

Inhaltsverzeichnis / Content

Vorwort	4
Foreword	4
Inhaltsverzeichnis / Content	5
Abkürzungen / Abbreviations	7
1 Ausgangslage und Übersicht.....	8
1 Initial Situation and Overview	9
2 Funktionale Dienstbeschreibung	10
2.1 <i>Einbettung in die IBIS-IP System Architektur des Fahrzeugs</i>	10
2.2 <i>Funktionen und Aufgaben des Dienstes</i>	10
2.3 <i>Funktionaler Ablauf</i>	11
2 Functional service description	12
2.1 <i>Embedding in the IBIS-IP System architecture of the vehicle</i>	12
2.2 <i>Description of Ticket Validation Service</i>	12
2.3 <i>Functional sequence</i>	13
3 Datenstruktur der Operationen / data structure of operations	14
3.1 <i>GetCurrentTariffStop</i>	14
3.1.1 Request	14
3.1.2 Response	14
3.2 <i>SubscribeCurrentTariffStop</i>	14
3.3 <i>UnsubscribeCurrentTariffStop</i>	14
3.4 <i>GetRazzia</i>	14
3.4.1 Request	14
3.4.2 Response	15
3.5 <i>SubscribeRazzia</i>	15
3.6 <i>UnsubscribeRazzia</i>	15
3.7 <i>GetCurrentLine</i>	15
3.7.1 Request	15
3.7.2 Response	15
3.8 <i>SubscribeCurrentLine</i>	16
3.9 <i>UnsubscribeCurrentLine</i>	16
3.10 <i>GetVehicleData</i>	16
3.10.1 Request	16
3.10.2 Response	16
3.11 <i>SubscribeVehicleData</i>	16

3.12	<i>UnsubscribeVehicleData</i>	17
3.13	<i>RetrieveTripData</i>	17
3.13.1	Request	17
3.13.2	Response	17
4	Versionshistorie/Version history	18
4.1	<i>Version 2.2</i>	18
4.1.1	Funktionale Erweiterungen Functional Upgrade	18
4.1.2	Technische Ergänzungen/Korrekturen Technical Upgrade/Corrections	18
4.2	<i>Version 2.3</i>	18
4.2.1	Technische Ergänzungen/Korrekturen Technical Upgrade/Corrections	18
	Tabellenverzeichnis / List of tables	19
	Regelwerke – Normen und Empfehlungen / References	20
	Impressum	21

Abkürzungen / Abbreviations

Die bereits in der VDV 301-1 definierten Abkürzungen werden an dieser Stelle nicht wiederholt.

The abbreviations already defined in VDV 301-1 are not repeated here.

1 Ausgangslage und Übersicht

Bislang fehlt in IBIS-IP eine Beschreibung, wie Fahrschein-Entwerter im Fahrzeug über IBIS-IP angesteuert werden.

Bisher wurden Entwerter über den IBIS Bus (VDV 300) angesteuert. Da der IBIS Bus in den Fahrzeugen zunehmend durch moderne Ethernet Verkabelung ersetzt wird, muss es möglich sein, die Entwerter auch über IBIS-IP ansteuern zu können.

Der Dienst TicketValidationService stellt dafür die erforderlichen Funktionen / Operationen bereit.

Folgende Anforderungen müssen durch den TicketValidationService erfüllt werden:

- Entwerter müssen die aktuelle Haltestelle und Zone auf den Fahrschein drucken können
- Entwerter müssen die aktuelle Linie drucken können
- Entwerter müssen die Fahrzeugnummer drucken können
- Entwerter müssen die aktuelle Fahrtrichtung drucken können
- Entwerter müssen ihre Funktion sperren können („Razzia“-Funktion)
- es müssen mehrere Entwerter in einem Fahrzeug unterstützt werden.

In der Regel wird der Dienst TicketValidationService vom Bordrechner des Fahrzeugs bereitgestellt.

1 Initial Situation and Overview

Up to now the VDV 301 specification does not contain a description how to handle ticket validator devices. The document at hand describes the necessary service for this task. Currently validators are controlled via the IBIS bus (VDV 300). As the IBIS bus is replaced with modern Ethernet cabling, it must be possible to control the validators via IBIS-IP. The service TicketValidationService provides the necessary functions and operation for this task.

Following requirements must be fulfilled by the TicketValidationService:

- Validator shall print the current stop and tariff area on a ticket
- Validator shall print the current line on a ticket
- Validator shall print the vehicle number on a ticket
- Validator shall print the current direction on a ticket
- On request validator shall be able to deactivate its function („Razzia“-functionality)
- It shall be possible to control several validators in a vehicle.

Normally the service TicketValidationService is provided by the onboard unit of the vehicle.

2 Funktionale Dienstbeschreibung

2.1 Einbettung in die IBIS-IP System Architektur des Fahrzeugs

TicketValidationService ist ein Dienst, der im IBIS-IP Netzwerk alle nötigen Informationen zum Ansteuern der Entwerter bereitstellt (in der Regel vom Fahrzeugrechner). Der Dienst wird von den Entwertern benutzt, um die Fahrscheine mit den relevanten Daten zu bedrucken.

Jeder Entwerter muss den DeviceManagementService anbieten, um das Gerät überwachen und steuern zu können (s. VDV 301-2).

Da es mehrere Entwerter im Fahrzeug geben kann, müssen diese über Ihre eindeutige Geräte-ID unterscheidbar sein. Die Geräte-ID kann z.B. durch Hersteller spezifische Parameter Files, Kodierstecker oder über den DeviceManagementService (Operation SetDeviceConfiguration, s. VDV-301-2) eingestellt werden. Dies wird in den meisten Fällen einmal bei der Inbetriebnahme des Fahrzeugs geschehen.

2.2 Funktionen und Aufgaben des Dienstes

Der TicketValidationService ist als Dienst gemäß VDV 301-2 implementiert und kann via DNS-SD entdeckt und via HTTP genutzt werden.

Der Dienst TicketValidationService stellt Operationen zur Verfügung, über welche der Entwerter alle benötigten Informationen beziehen kann. Bei jedem Haltestellenwechsel und bei Kontrollen (Razzia aktiviert) werden die entsprechende Werte übertragen.

Übersicht der bereitgestellten Operationen:

- **GetCurrentStopPoint** - liefert die Informationen zur aktuellen Haltestelle
- **SubscribeCurrentStopPoint** - abonniert die Informationen zur aktuellen Haltestelle
- **UnsubscribeCurrentStopPoint** - zugehörige Unsubscribe Operation
- **GetRazzia** - gibt an, ob Razzia gesetzt wurde
- **SubscribeRazzia** - abonniert, ob Razzia gesetzt wurde
- **UnsubscribeRazzia** - zugehörige Unsubscribe Operation
- **GetCurrentLine** - gibt die Informationen zur aktuellen Line
- **SubscribeCurrentLine** - abonniert die Information zur aktuellen Line
- **UnsubscribeCurrentLine** - zugehörige Unsubscribe Operation
- **GetVehicleData** - gibt die Informationen zum Fahrzeug
- **SubscribeVehicleData** - abonniert die Information zum Fahrzeug
- **UnsubscribeVehicleData** - zugehörige Unsubscribe Operation
- **RetrieveTripData** - gibt die für den Haltepunkt definierten Fahrtinformationen zurück

Datum und Zeit bezieht der Entwerter über den Dienst TimeService (s. VDV 301-2-10).

2.3 Funktionaler Ablauf

Je nach Projektanforderung benötigt der Entwerter unterschiedliche Daten für den Fahrschein Aufdruck. Im einfachsten Anwendungsfall muss der Entwerter nur die Haltestelle, Zone und Datum / Uhrzeit aufdrucken. Für dieses Anwendungsbeispiel werden nur wenige Operationen benötigt und der Ablauf würde wie folgt aussehen:

Ein Entwerter ist nach seinem Start (Power on) solange gesperrt, bis er erstmalig Daten zur Haltestelle, Zone, Datum und Zeit erhält. Beim Start abonniert der Entwerter die beiden Services „SubscribeCurrentStop“ und „SubscribeRazzia“ und ermittelt über den TimeService, wo er das aktuellen Datum und Uhrzeit beziehen kann. Sobald das Fahrzeug eine Haltestelle erreicht, wird die Haltestelle und Zone dem Entwerter mitgeteilt.

Wird die Razzia Funktion aktiviert / deaktiviert (z.B. durch den Fahrer am MMI des Bordrechners), wird dies den Entwertern mitgeteilt, um die Druckfunktion zu sperren / wieder frei zu geben.

2 Functional service description

2.1 Embedding in the IBIS-IP System architecture of the vehicle

TicketValidationService is a service, which provides all necessary information in a IBIS-IP network to control validators. Normally this service is provided by the onboard unit. A validator uses this service to print the required data on a ticket.

Each validator must offer the DeviceManagement Service to monitor and control the device (s. VDV 301-2).

As a vehicle contains several validators, the validators must be distinguishable by an unique device Id. The device Id can be set e.g. via vendor specific parameter files, coded plug or over the DeviceManagementService (Operation SetDeviceConfiguration, s. VDV-301-2). In most cases the device ID is set during the setup / installation of the vehicle.

2.2 Description of Ticket Validation Service

The TicketValidationService is realized according VDV 301-2 and can be recognized via DNS-SD and can be used via http.

The service TicketValidationService provides operation, which deliver the necessary information for the validator. The information of each stop change and also in case of ticket checks (Razzia activated) are transferred to the validator.

Overview of the supported operations:

- **GetCurrentStopPoint** provide information of current stop
- **GetRazzia** provide, if Razzia was activated
- **SubscribeCurrentStopPoint** subscribe information of current stop
- **UnsubscribeCurrentStopPoint** related unsubscribe operation
- **SubscribeRazzia** subscribe, if Razzia was set
- **UnsubscribeRazzia** related unsubscribe operation
- **GetCurrentLine** provide information of current line
- **SubscribeCurrentLine** subscribe information of current line
- **UnsubscribeCurrentLine** related unsubscribe operation
- **GetVehicleData** provide information of the vehicle
- **SubscribeVehicleData** subscribe information of the vehicle
- **UnsubscribeVehicleData** related unsubscribe operation
- **RetrieveTripData** deliver for a stop the defined trip information

Date and time is delivered to the validator over the service TimeService (s. VDV 301-2-10).

2.3 Functional sequence

Depending on the project requirements the validator needs different information to generate the print on the ticket.

In the simplest case the validator must only print stop, zone and date / time on the ticket. In this case only few operations are needed and the proceeding is the following:

After start up a validator is blocked until the validator get data of a stop, zone, date and time the first time. During startup the validator subscribes both services „SubscribeCurrentStop“ and „SubscribeRazzia“ and determines over the TimeService, where the current date and time can be received. As soon as the vehicle reaches a stop the validator is informed about the stop and zone.

If the razzia (ticket check) function is activated / deactivated (e.g. driver is pressing the button on on-board unit MMI) the validator is informed to block / unblock the print functionality.

3 Datenstruktur der Operationen / data structure of operations

This chapter describes the operations and their data structure. For the implementation, the corresponding XSD file shall be used, in case of mismatches between the description below and the XSD the XSD is the master.

3.1 GetCurrentTariffStop

3.1.1 Request

A get operation has no RequestStruktur.

3.1.2 Response

TicketValidationService.GetCurrentStopPointResponse			+Structure	
			Choice	
a	CurrentTariffStopData	-1:1	+TicketValidationService.CurrentTariffStopData	See below
b	OperationErrorMessage		IBIS-IP.string	Error response

Table 1 Description of TicketValidationService.GetCurrentStopPointResponse

TicketValidationService.CurrentTariffStopData			+Structure	
	TimeStamp	1:1	IBIS-IP.dateTime	Time stamp of the answer
	CurrentTariffStop	1:1	+StopInformation	Information of current stop See (2) chapter 2.50
	CurrentTripRef	0:1	IBIS-IP.NMToken	Reference of current trip

Table 2 Description of TicketValidationService.CurrentStopPointData

3.2 SubscribeCurrentTariffStop

To establish the subscription the data structure described in (2) chapter 2.54 is used.

3.3 UnsubscribeCurrentTariffStop

To terminate the subscription the data structure described in (2) chapter 2.60 is used.

3.4 GetRazzia

3.4.1 Request

A get operation has no RequestStruktur.

3.4.2 Response

TicketValidationService.GetRazziaResponse			+Structure	
			Choice	
a	RazziaData	-1:1	+TicketValidationService.RazziaResponseData	See below
b	OperationErrorMessage		IBIS-IP.string	Error response

Table 3 Description of TicketValidationService.GetRazziaResponse

TicketValidationService.RazziaResponseData			+Structure	
	TimeStamp	1:1	IBIS-IP.dateTime	Time stamp of the answer
	RazziaState	1:1	TicketRazziaInformationEnumeration	Razzia activated or not See (2) chapter 3.24

Table 4 Description of GetrazziaResponsetData

3.5 SubscribeRazzia

To establish the subscription the data structure described in (2) chapter 2.54 is used.

3.6 UnsubscribeRazzia

To terminate the subscription the data structure described in (2) chapter 2.60 is used.

3.7 GetCurrentLine

3.7.1 Request

A get operation has no RequestStruktur.

3.7.2 Response

TicketValidationService.GetCurrentLineResponse			+Structure	
			Choice	
a	CurrentLineData	-1:1	+TicketValidationServiceCurrentLineData	See below
b	OperationErrorMessage		IBIS-IP.string	Error Response

Table 5 Description of TicketValidationService.GetCurrentLineResponse

TicketValidationService.CurrentLineData			+Structure	
	TimeStamp	1:1	IBIS-IP.dateTime	Time stamp of answer
	CurrentLine	1:1	+LineInformation	Information of line See (2) chapter 2.31

Table 6 Description of TicketValidationService.CurrentLineData

3.8 SubscribeCurrentLine

To establish the subscription the data structure described in (2) chapter 2.54 is used.

3.9 UnsubscribeCurrentLine

To terminate the subscription the data structure described in (2) chapter 2.60 is used.

3.10 GetVehicleData

3.10.1 Request

A get operation has no RequestStruktur.

3.10.2 Response

TicketValidationService.GetVehicleDataResponse			+Structure	
			Choice	
a	VehicleData	-1:1	+TicketValidationService.VehicleDat a	See below
b	OperationErrorMessage		IBIS-IP.string	Error response

Table 7 Description of TicketValidationService.GetVehicleDataResponse

TicketValidationService.VehicleData			+Structure	
	TimeStamp	1:1	IBIS-IP.dateTime	Time stamp of answer
	VehicleRef	1:1	IBIS-IP.NMTOKEN	Reference of vehicle-ID
	RouteDeviation	0:1	RouteDirectionEnumeration	Information , if deviation from line route exist See (2) chapter 3.20
	DoorOpenState	0:1	DoorOpenStateEnumeration	Information about the door state See (2) chapter 3.9
	MovingDirectionForward	0:1	IBIS-IP.boolean	Information about the direction
	VehicleMode	0:1	VehicleModeEnumeration	Information about vehicle typ See (2) chapter 3.26
	DriverNumber	0:1	IBIS-IP.string	Number of the signed on driver

Table 8 Description of TicketValidationService.VehicleData

3.11 SubscribeVehicleData

To establish the subscription the data structure described in (2) chapter 2.54 is used.

3.12 UnsubscribeVehicleData

To terminate the subscription the data structure described in (2) chapter 2.60 is used.

3.13 RetrieveTripData

3.13.1 Request

TicketValidationService.RetrieveTripDataRequest			+Structure	
	TripRef	1:1	IBIS-IP.NMTOKEN	Reference of the requested trip

Table 9 Description of TicketValidationService.RetrieveTripDataRequest

3.13.2 Response

TicketValidationService.RetrieveTripDataResponse			+Structure	
			Choice	
a	TripData	-1:1	+TicketValidationService.TripData	See below
b	OperationErrorMessage		IBIS-IP.string	Error Response

Table 10 Description of TicketValidationService.RetrieveTripDataResponse

TicketValidationService.TripData			+Structure	
	TimeStamp	1:1	IBIS-IP.dateTime	Time stamp of answer
	TripInformation	1:1	+TripInformationStructure	See (2) chapter 2.57

Table 11 Description of TicketValidationService.TripData

4 Versionshistorie/Version history

4.1 Version 2.2

4.1.1 Funktionale Erweiterungen Functional Upgrade

Keine/none

4.1.2 Technische Ergänzungen/Korrekturen Technical Upgrade/Corrections

- CurrentStopPoint is used in CustomerInformationService and TicketValidationService, but with a different meaning in both services. So CurrentStopPoint is renamed to CurrentTariffStop, because this value is changed when the next bus stop has been reached.

4.2 Version 2.3

4.2.1 Technische Ergänzungen/Korrekturen Technical Upgrade/Corrections

- Correction of chapter 3.1.2, description is now in line with XSD, i.e. no XSD update is necessary.

Tabellenverzeichnis / List of tables

Table 1	Description of TicketValidationService.GetCurrentStopPointResponse	14
Table 2	Description of TicketValidationService.CurrentStopPointData	14
Table 3	Description of TicketValidationService.GetRazziaResponse	15
Table 4	Description of GetrazziaResponsetData	15
Table 5	Description of TicketValidationService.GetCurrentLineResponse	15
Table 6	Description of TicketValidationService.CurrentLineData	15
Table 7	Description of TicketValidationService.GetVehicleDataResponse	16
Table 8	Description of TicketValidationService.VehicleData	16
Table 9	Description of TicketValidationService.RetrieveTripDataRequest	17
Table 10	Description of TicketValidationService.RetrieveTripDataResponse	17

Regelwerke – Normen und Empfehlungen / References

- (1) VDV 301-2-0 IBIS-IP Beschreibung der Dienste / Service description
Basisdienste / Base Services
DeviceManagementService, SystemManagementService,
SystemDocumentationService V2.0, 02/2018
- (2) VDV 301-2-1 IBIS-IP Beschreibung der Dienste / Service description
Gemeinsame Datenstrukturen und Aufzählungstypen/
Common data structures and enumerations, 05/2017
- (3) VDV 301-1 IBIS-IP- Teil 1: Systemarchitektur / System architecture
V1.0, 01/2014
- (4) EN 13149- Öffentlicher Verkehr - Planungs- und Steuerungssysteme für
Straßenfahrzeuge - Teil 8: Physikalische Schicht für IP-Kommunikation
Englische Fassung CEN/TS 13149-8:201

Die IBIS-IP XSD-Dateien stehen unter www.vdv.de/ip-kom-oev.aspx zum Download bereit.

The IBIS-IP XSD files are available for download at www.vdv.de/ip-kom-oev.aspx.

Impressum

Verband Deutscher Verkehrsunternehmen e. V. (VDV)
Kamekestraße 37-39 · 50672 Köln
T 0221 57979-0 · F 0221 57979-8000
info@vdv.de · www.vdv.de

Ansprechpartner

Dipl.-Ing. Berthold Radermacher
T 0221 57979-141
F 0221 57979-8141
radermacher@vdv.de

Verband Deutscher Verkehrsunternehmen e. V. (VDV)
Kamekestraße 37-39 · 50672 Köln
T 0221 57979-0 · F 0221 57979-8000
info@vdv.de · www.vdv.de
