Recommendation 452

7/13

VDV Data Model 5.0 - "Interface Initiative"

Standard VDV Route Network / Timetable Interface

including enhancements:

- Transfer definitions including their validity
- Administrative units

Version: 1.5

Editing:

Committee for Information Processing (AIV)

Processing:

VDV Data Model Task Force

Verband Deutscher Verkehrsunternehmen (VDV)

Kamekestraße 37 - 39, 50672 Cologne, Tel: +49 221/57979-0, Fax: +49 221 514272

Members of the "VDV Data Model" task force when this document was prepared:

Distribution list:

Uwe Amann, Dipl.-Ing, ELGEBA Gerätebau GmbH, Bad Honnef
Peter Arnsberger, Mentz Datenverarbeitung GmbH (MDV), Munich
Georg Aschenbrenner, Ing, Zelisko GmbH Fabrik für Elektrotechnik und Mödling
Arne Baudis, Dipl.-Ing, BBR Verkehrstechnik GmbH, Braunschweig
Michael Beck, Dipl.-Inform, initplan GmbH, Karlsruhe
Stefan Bessler, Stadtwerke München GmbH, Munich
Wilfried Düx, Mentz Datenverarbeitung GmbH (MDV), Munich
Christian Duisberg, PSI Transcom GmbH, Berlin
Gerald Dury, Dipl.-Inform. (FH), Trapeze Switzerland GmbH, Neuhausen Rhine Falls
Frithjof Eckardt, Eckardt Software Management GmbH, Hannover
Martin Finker, MOVEO Software GmbH, Potsdam
Thorsten Fischer, Telenet AG, Darmstadt
Linus Gebhardt, TMG TeleMatrik GmbH, Lauterbach
Thomas Giemula, Dipl.-Ing, Bundesministerium für Verkehr, Bau und Stadtentwicklung, Berlin
Michael Greiner, DILAX Intelcom AG, Ermatingen
Andreas Gruhl, ISIDATA GmbH, Hannover
Joachim Haas, IVU Traffic Technologies AG, Berlin
Kai Horn, Zelisko GmbH Fabrik für Elektrotechnik und Mödling
Karl-Friedrich Hüssner, Dipl.-Ing, DILAX Intelcom AG, Ermatingen
Jörg Hutschenreiter, Dr. Dipl.-Math, CSC Deutschland Solutions GmbH, Dresden
Rolf Kerger, DB Mobility Logistics AG, Frankfurt
Volkmar Klos, Unternehmensberatung für Verkehr und Technik GmbH (UVT), Mainz
A. Knapp, Lawo-MARK IV Industries GmbH, Rastatt
Oliver Koch, IVU Traffic Technologies AG, Berlin
Werner Kohl, Dipl.-Math, Mentz Datenverarbeitung GmbH (MDV), Munich
A. Krampe, CSC Deutschland Solutions GmbH, Dresden
Karl Horst Lenzen, T-Systems GEI GmbH BU Geschäftsstelle Hamburg, Mülheim
Bettina Lisbach, init Innovative Informatikanwendungen in Transport-, Verkehrs- u. Leitsyst. GmbH, Karlsruhe
Altomaro Locurcio, Dipl.-Ing, AL-Engineering AG, Dillingen
Patrik Lorenc, Dipl.-Ing, Unternehmensberatung für Verkehr und Technik GmbH (UVT), Mainz
Sven Matz, W&W Systemberatung GmbH -turista-,
Achim Müller, Dipl.-Ing, Ingenieurbüro STAM System Technik Achim Müller, Buxtehude
Dr. Klaus Nökel, PTV AG Karlsruhe Hauptsitz, Karlsruhe
Fridolin Ritter, Dipl.-Inform, COS Gesellschaft für Computersysteme und Organisation, Oberkirch
René Rothe, Trapeze Germany GmbH, Berlin

Frank Schenkenberger, initplan GmbH, Karlsruhe

Johann Schmickl, Dipl.-Wirtsch.-Ing, Stuttgarter Straßenbahnen AG, Stuttgart

Ute Siaden Ortega, Eckardt Software Management GmbH, Hannover

Josef Spinner, COS Gesellschaft für Computersysteme und Organisation, Oberkirch

Joachim Srocka, IVU Traffic Technologies AG, Berlin

Rainer Stimmerling, Verkehrsautomatisierung Berlin GmbH, Berlin

Gustav Thiesing, Dipl.-Ing, Beratungsgesellschaft für Leit- und Informationssystemen, Braunschweig

R. Zohar, init Innovative Informatikanwendungen in Transport-, Verkehrs- u. Leitsyst. GmbH, Karlsruhe

Table of Contents

HISTORY

Version 1.3 has been enhanced compared with the previous version and now includes "Transfer data for AVLC". This allows information concerning planned transfers to be transferred from a planning system to an AVLC.

In **Version 1.4** intermediate points were introduced in 'point_type' and coordinates in 'stop' to support a graphical representation.

Version 1.5 now includes the modelling of zones, references to the new NeTEx European standard and extensions in the area of journey connection.

Date	**Author**	**Change**	**New version**
16/11/2004	W. Bruns	New in chapter 10: Extension of the "Transfer protection" function Additions to chapters 5.1 and 8.2 concerning transfer data	1.3
		With the creation of Version 1.2, the file codes for the BASIS_VER_GUELTIGKEIT and BASE_VERSION (MENGE_BASIS_VERSIONEN) tables have been exchanged. Incorrect: BASE_VERSION_VALID (BASIS_VER_GUELTIGKEIT) 485 BASE_VERSION (MENGE_BASIS_VERSIONEN) 993 Correction: BASE_VERSION_VALID (BASIS_VER_GUELTIGKEIT) 993 BASE_VERSION (MENGE_BASIS_VERSIONEN) 485	
02/03/2005	W. Bruns	Correction: FEEDER_STOP_NO *(ZUB_ORT_REF_ORT)* to data type "decimal (6)"	
08/03/2005	W. Bruns	Extension of the forecast with TRIP_TYPE_NO (FAHRT_TYP_NR)	
14/03/2005	W. Bruns	Introduced: **PSI*traffic*/AVLC**	
24/06/2005	W. Bruns	Correction of the introduction to chapter 10: RouteID (LinienID), DirectionID (RichtungsID)	1.4
09/10/2007	W. Bruns	Incorporation of the value range changes 1 to 5 and extensions to STOP (REC_ORT) from Appendix 12	1.4

29/10/2007	Christian Rossol, PTV AG	Reproduction of the geographic sequence of the network patterns. The new 'intermediate pattern point' in the 'POINT_TYPE' (MENGE_ONR_TYP) table allows the unique differentiation from the existing location markers, thereby avoiding any potential interpretation problems. The coordinate information is taken from the STOP (REC_ORT) table. In detail: 45: Extension of the table number for the 'JOURNEY_TYPES' (MENGE_FAHRTART) table Page 31: Extension of the POINT_TYPE (MENGE_ONR_TYP) table with the POINT_TYPE (ONR_TYP_NR) type 5 = intermediate pattern point; see page 55: Extension of the validity for POINT_TYPE (ONR_TYP_NR) of the STOP (REC_ORT) table up to the value "5". Page 50: Extension of the "POINT_ON_LINK" (REC_SEL_ZP) table with the "POINT_ON_LINK_SEQ_ID" (ZP_LFD_ID) column.	
04/12/2007	Gerald Dury	The DIRECTION (ORT_RICHTUNG) attribute has been added to STOP (REC_ORT), which describes the direction of entry into the stopping point. This is an important attribute for the GPS location, which is also transferred from the planning systems to the AVLC systems. Extension of the value range of the zone area to 1-99,999 in accordance with the modification document in the Internet.	
10/12/2007	Winfried Bruns	Transmodel equivalent	
19/05/2008	Winfried Bruns	English term for TRAVEL_TIME (UEB_FAHRZEIT) in DEAD_RUN_TIME (UEB_FZT) corrected to TO_POINT_NO	1.4a
10/2009		Introduced: ROUTE_SEQUENCE (LID_VERLAUF) table, REQUEST_STOP (BEDARFSHALT) attribute	
03/04/2013	Winfried Bruns	Additional attributes for ROUTE_SEQUENCE (LID_VERLAUF), STOP (REC_ORT), JOURNEY (REC_FRT)	1.5
03/04/2013	Winfried Bruns	Additional attributes for JOURNEY CONNECTION (EINZELANSCHLUSS)	1.5
03/04/2013	Winfried Bruns	Extension of the zones	1.5
23/04/2013	Winfried Bruns	Introduction of a through-running service in JOURNEY (REC_FRT)	1.5
09/05/2013	Winfried Bruns	'Productive' attribute added, in accordance with comments from Telko on 26.6.13	1.5
2019/05/02	WB, GD	Addition LinienID in REC_LID, DFID in REC_FRT, TLID	1.5 inkl. CRs2019
2019/06/06	Wb	Chapter 9.1.3 deleted.	

The core of this VDV Recommendation remains unchanged from Version 1 published in 1999. Changes are essentially restricted to value ranges and minor extensions of the structures, which should not affect the downwards compatibility of the interface.

1 Foreword to the VDV Data Model 5.0:

1.1 The VDV Data Model as a Reference Point for Data Modelling in Public Transport

After the first publication in 1990 of the “VDV Data Model” Recommendation, it has become the basis for data modelling in public transport. Many system suppliers have followed the VDV Data Model and used it as a basis for developing their own product data model.
Perhaps it was due to the great success of the VDV Data Model that the VDV faced increasing demands to develop practically orientated solutions, extending even beyond the capabilities of the VDV Data Model. Ideas included standard interfaces which, thanks to their plug-in compatibility, are instantly usable, and which permit standard software modules to communicate with each other at a reasonable cost.

1.2 The VDV Interface Initiative VDV Data Model

That is why in 1998 the VDV decided to establish an initiative entitled “The VDV Interface Initiative” in order to promote the creation of standardised data interfaces based on the VDV Data Model.

These interfaces basically represent a part of the VDV Data Model. We are not therefore dealing with a new concept, but with a logical application of the VDV Data Model which was the result of many years of investigation. Provision of a more exact description and an expansion of the technical specifications concerning data transfer, as well as functional aspects, means, however, that it is more practice-oriented than was the case with the simple VDV Data Model.

This current edition of the VDV Recommendation contains the first interface definition from the initiative. It deals with the **“Network and Timetable”** area. The definition distinguishes itself from the VDV Data Model insofar as it has the following characteristics:

- In conjunction with SQL access as required in earlier versions of the VDV Data Model, an alternative file format is defined for off-line data transfer (see VDV Recommendation 451)
- The minimum scope of the data model is clearly outlined
- The range of values is more restrictively defined for the individual attributes (from the user's perspective).
- The individual attributes have been described in more detail and therefore more precisely.

2 Definitions

Standard VDV Route Network / Timetable Interface

An interface definition based on the VDV Data Model for the transfer of network and timetable data. It consists of a definition of the data model and the two possibilities for gaining access – SQL and VDV file format.

VDV Database

Relational database based on the VDV Data Model. The used section focuses on the data model for the VDV standard interfaces. The VDV database can form part of a product-specific database. Data can be transferred into and out of the VDV database using SQL or VDV file format.

VDV File Format

Qualified ASCII data format for the off-line data transfer of specified VDV Data Model data.

VDV Standard Interfaces Compatibility

A software system is regarded as being compatible when it is capable of exporting data into the VDV database or importing data from it. It does not matter whether this occurs using files in VDV file format or via direct SQL access to the VDV database. In both cases, the functions and consistency tests as described in section 5.5 must be adhered to. If there are any discrepancies between the content of this recommendation and the “VDV Data Model” paper (especially as regards attribute value range), this one should be viewed as a continuation. Therefore, the information in this document is decisive.

Planning Program

Software for vehicle and crew scheduling in public transport systems.

AVLC

The Automatic Vehicle Location and Control System (AVLC) is the new term for automated control systems, which takes into account the increased capacity of these systems.

3 Objectives

In the field of public transport, various manufacturers' software modules are used. Data exchange between these software modules is frequently necessary. Various departments within the transport industry and also the general public need up-to-date timetable data which is drawn up by traffic planning. For example, it is required for:

- Transit operations' supervision and control with an AVLC
- Statistics
- Passenger counting
- Counting of handicapped passengers
- Crew scheduling and personnel arrangements
- Dynamic passenger information
- Timetable information
- Depot management systems

Establishing such information flows is a very expensive procedure, especially when specific interfaces have to be written in each individual case.

The standardisation of interfaces for the exchange of data between public transport software systems as part of the "VDV Data Model Interface Initiative" therefore pursues the following aims:

- General minimisation of individual interfaces
- Avoidance of repeated updating
- Provision of a more exact description and an expansion of technical specifications concerning data transfer, as well as functional aspects, means, however, that it is more practice oriented than was the case with the simple VDV Data Model.
- Interfaces which function independently of the systems involved
- Use of the same interface for each transport company (standard product)
- Transparency of data for all systems
- Important numerical or alphanumerical data fields (key attributes) are identically allocated in both systems
- Uniform requirements list of the transport authorities

4 Limitations

The “VDV Standard Interface Network / Timetable” interface description is exclusively concerned with data describing networks and timetables. It therefore represents a section of the VDV Data Model V.4.1.

The VDV Data Model 5.0 concentrates exclusively on the data structures of interfaces between software modules in public transport systems. The individual internal data structures of the systems are not part of this specification and are also (contrary to the earlier versions of the VDV data model) omitted from any compatibility tests.

In many cases it will still make sense to compare the proprietary data model with the VDV Data Model.

5 General Description

The aim of the **"Standard VDV Route Network / Timetable Interface"** is to transfer network definitions and timetables from a source system into a target system. As a general rule, the timetable data from a (vehicle and crew) scheduling programme is passed on to the consumer systems for the purpose of operation monitoring and control (AVLC), cost control and/or publication.

When transferring data from a planning system into an AVLC, the data in the AVLC can be supplemented by the user with AVLC-specific data.

This will be referred to as "AVLC-specific data". Examples of data which are updated in the AVLC and which are not mapped in the **VDV Standard Interface "Network / Timetable"**:

- Traffic light influencing parameters
- Radio parameterisation for the AVLC
- Data for dynamic passenger information
  - Stop-related additional information
  - Free texts
  - Differentiation of a planned/actual comparison or combination

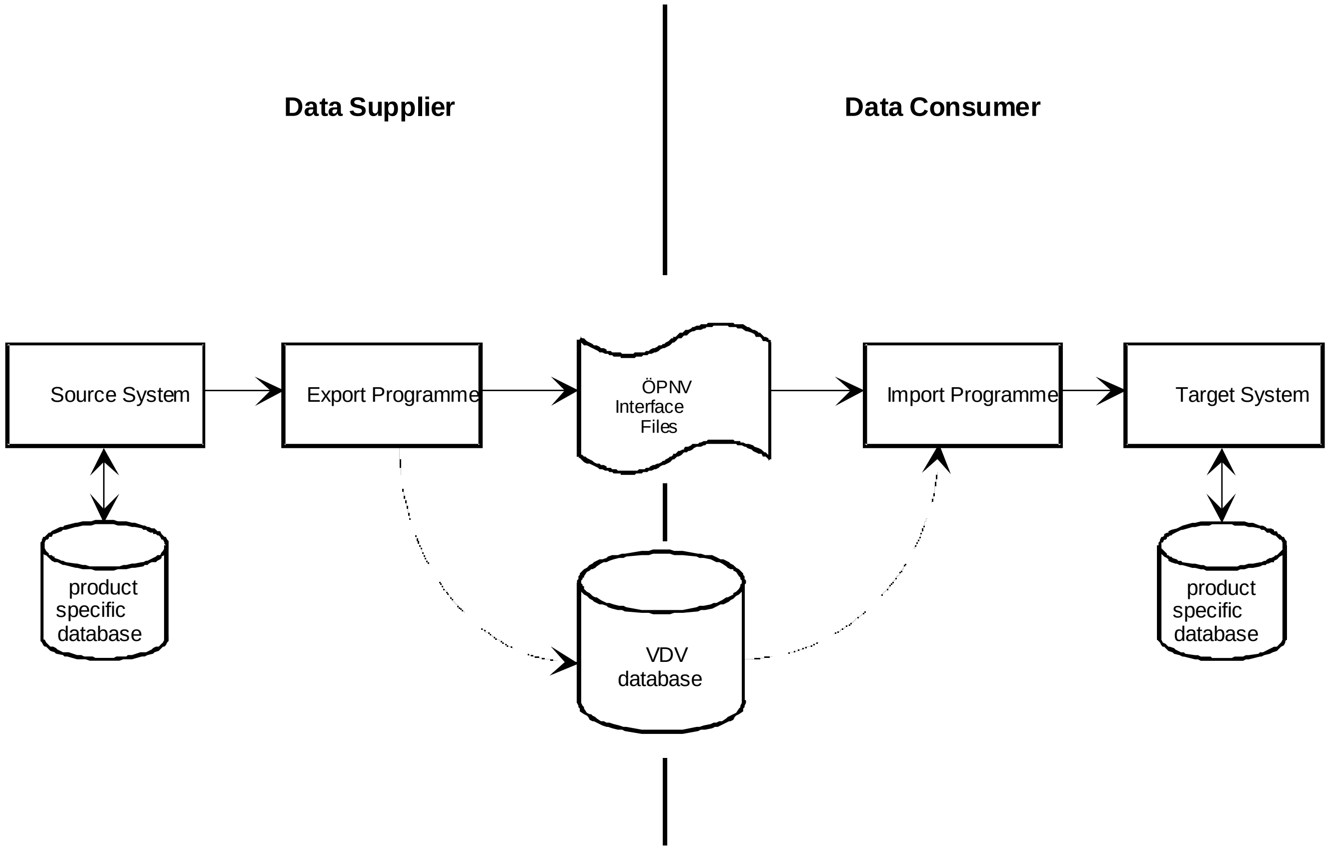
With a renewed data transfer from the **Standard VDV Route Network / Timetable Interface**, the AVLC-specific data, which is already included in the AVLC database, must be taken into consideration.

5.1 Scope of Data

The **Standard VDV Route Network / Timetable Interface** comprises the following data:

- Calendar data (day types and their validity in the period)
- Operational data (vehicle stock, vehicle types, announcement texts and destination texts)
- Location data (bus stops, stopping points, beacons, depots)
- Network data (route sections, distances, travel time groups, travel times, stopping times)
- Route data (routes and courses for different patterns)
- Timetable data (runs and run-dependent stopping times, blocks)
- In Version 1.3 (chapter 10) transfer data has been included in the interface, which facilitates the transfer of journey connection definitions together with their validity for example from a journey planning system to an AVLC, thus laying the foundation for the protection of and information concerning transfers.
- In Version 1.5 (chapter 11) it is now possible to use zones or administrative units.

5.2 Data flow


Data Supplier
Data Consumer
Source System
Export Programme
ÖPNV Interface Files
Import Programme
Target System
product specific database
VDV database
product specific database

5.3 Interface files

Data exchange using interface files becomes necessary under the following circumstances:

- Data is imported or exported from an external system. The data may possibly be reused on another hardware platform.
- Data must be post processed, inspected or evaluated using standard market software, for example:
  - a database is to be inspected or modified using a text editor,
  - a database is to be imported or exported (possibly with the aid of additional macros) using a spreadsheet programme.

5.4 SQL Access

Access to **VDV Standard Interface "Network / Timetable"** data should also be possible via an SQL interface which enables direct (interactive) access to the VDV database. This means that data in the VDV Database can be modified, deleted or selectively downloaded.

5.5 Requirements on the files

Formal Conditions

- The data structure (tables, attributes, value ranges) corresponds to the description published in this recommendation.
- Data transfer takes place via VDV interface files or via SQL access
- The integrity of the references for the network and timetable data must be guaranteed by the exporting system.
- The consistency and completeness of the database must be checked by the system which is exporting the data.

Logical and content-related conditions

A prerequisite for the successful implementation of the interface is that the logical and content-related relationships of the network and timetable data have been correctly mapped. This includes, for example, that

- the departure times of successive trips can be upheld on the basis of the underlying route definitions,
- the data elements are uniquely identifiable (e.g. unique stop codes, route numbers, pattern numbers per route, run numbers per route, block numbers).
- blocks have unbroken coverage, beginning with exit from the depot as far as return into the depot

The logical and content-related conditions are already guaranteed by the data supplier when the data is exported.

Individual conditions will be dealt with in the specifications (further below)

Apart from the conditions described in this recommendation issue, **company conditions** must also be met if export of data for an AVLC is planned.

Example: Data transfer from the Interplan transport planning system to LIO-Data in order to update the LIO AVLC.

- Some planning systems only record those runs which are operated by a transport company on a productive level (e.g. for the production of timetable information). In order to achieve an exact model of operational activity and successfully update all the AVLC components, all runs should in fact be recorded in the planning system.
- If a transport authority uses trams, turning loops are often employed at the terminal stops of these routes. If the turning loop data (e.g. distances and stopping points) is not transferred from Interplan to LIO-Data via the VDV-Import function, this path data is missing for the relevant AVLC components and must be manually recorded in LIO-Data. This leads to a significantly increased data management workload in LIO-Data, which could be avoided by completing the route and timetable data accordingly in Interplan. The "VDV-Import-Interface" only accepts the data that is made available to it, which it then converts for the LIO operations control system. It does not make any changes at the pattern or timetable level.
- The distances between the stops must be measured exactly and entered in the Interplan planning system (accurate to within one metre), as these measurements along with the beacons and GPS coordinates form the basis for the logical location process of the AVLC.
- The journey times between the bus stops and the stopping times should also be recorded as exactly as possible (in seconds), as the timetable theoretical versus actual comparison depends on these values. If the times are recorded to the nearest minute, then the timetable comparison cannot deliver exact results either at the control centre or on the on-board vehicle computer.
- The quality of the distances and times measured in the planning system has a direct effect on the operation of the AVLC. That is because this data forms the basis for the navigation, dynamic passenger information, transfer protection and statistics etc.

6 Application areas

6.1 Export Network / Timetable

The specification enables a data supplier to convert product-specific network and timetable data into a standardised format. An application for data export could be considered for:

- Journey scheduling programs (e.g. for supplying an AVLC system) or
- AVLC, for supplying a company database.

6.2 Import Network / Timetable

The specification enables the data consumer to convert standardised network and timetable data into product-specific data. An application for data import could be considered for:

- AVLC system (from vehicle and crew scheduling programs)
- Timetable information
- Ticket printer
- Passenger counting
- Company traffic database

...

6.3 Network / Timetable Data Exchange

A data transfer system based on the VDV Standard Route Network / Timetable Interface is notable for its controlled redundant database organisation. This means that the network and timetable data is only recorded and updated in a source system (e.g. in the scheduling program) and is transferred to the data consumer (e.g. AVLC) for further processing. The database in the target system therefore corresponds to a mapping of the data in the source system. The data consumers have their own database in their product-specific databases.

The data consumers (target systems) generally require further internal data for their operation to be productive. It cannot always be supplied by the source system (e.g. with an AVLC, the beacons and their position on the route) and must therefore be completed in the target system.

Data adjustment in the consumer system through data import

If new data is imported from the source system during a data transfer, then this new data must be compared with the data in the target system by the import program. This process can be performed by a so-called update function, which, when importing the data, re-uses the target system specific data as much as possible. The comparison between the source system interface data and the data which is already present in the target

system must be undertaken in a logical sequence. The data must first be read, then compared or completed and only then imported by the target system.
For example, beacons are provided along the route of an AVLC system. A new data transfer using an identical route and different run and stopping times should not affect these beacon positions.

Data Comparison in the Source System by Updating

When transferring data from a source system into the target system via interface files or SQL access, existing data is replaced by the new data. If, as an exception, the changes to the data are made in the target system directly, then you have to make sure, prior to the next data exchange, that the corresponding changes were also made in the source system. If no such updating occurs in the source system, then the changes to data in the consumer system will be overwritten when the next data exchange takes place.

7 Compatibility

Application software interfaces may be compatible with the interface described here. The following conditions must be met:

- The interface must use exactly the same data model as described in this publication.
- The data must be stored in VDV file format and / or in a relational database.
- The interface must be available to transport companies as a product of the provider.
- It must be applicable independently of individual customers.
- In order to achieve an exact model of operational activity and successfully update all the AVLC components, all runs should in fact be recorded in the planning system.

Compatibility only applies to the published version of the interface program. Any changes to the interface programme means it is necessary to carry out a new examination. On the request of the software provider, the test is carried out by the VDV. The “VDV-Import-Interface” only accepts the data made available to it and converts it for the LIO.

Depending on the eventual use, various types of compatibility with the interface description are possible:

- An **export compatibility** exists if the software’s own database can provide network and timetable data for another application.
  A certain minimum scope is required. The corresponding tables in the recommendation, listed in the "Minimum scope of the AVLC" column, are marked with a black square.

- The quality of the distances and times measured in the planning system has a direct effect on the operation of the AVLC.
- If data is going to be used in an AVLC, a **minimum scope** is required. (The corresponding tables in the recommendation, listed in the "Minimum scope of the AVLC" column, are marked with a black square). An interface which is capable of doing this is designated as **import compatible for AVLC**.
- **Full compatibility** comprises export compatibility and import compatibility for AVLC, i.e. data exchange in both directions.

Scope of the implemented interface

The specification to hand describes the minimum scope of an interface. Scheduling programs (e.g. to update an AVLC system) or an AVLC system e.g. to update a company database
Suggestions for the standard interface to expand to include further tables are welcome. The VDV will examine these and publish them in a subsequent issue of the recommendation.

8 Compatible Products

The official VDV website, WWW.VDV.DE/oepnv-datenmodel.aspx, lists and describes the products (interfaces) that are compatible with the Standard VDV Route Network / Timetable Interface. In some cases a compatible version of the interfaces is still nowhere in operation but has nevertheless been planned (depending on orders). Corresponding annotations have been made.

The use of these products is recommended by the VDV, since, by using them as a basis, the information flow between software applications in public transport systems is facilitated. This recommendation only refers to the capability of the software in question to export or import data via the VDV Standard Interface. No statement can be made here regarding the general quality of the software and especially regarding its ability to fulfil company requirements.

In order to achieve compatibility status, the software manufacturer had to:

- provide the VDV with a performance description of the interface including data model, product name and version and
- prove that the interface performed according to the description.

Details regarding locations for installation are pending further information from the software providers.

8.1 Application Matrix for Relations

A prerequisite for coupling two products is, in addition to being compatible with the interface description as published in this recommendation issue, that the source system is capable of delivering the relations which are required by the target system.

When data exchange takes place, basically all the tables contained in the Network / Timetable VDV Standard Interface are transferred. Depending on what products are involved, it is however possible that some tables will be transferred empty.

The tables on the official VDV website (WWW.VDV.DE/oepnv-datenmodel.aspx) show the relations that are supported by the various products.

For good coupling possibilities to exist, it is generally very important to have **the largest possible number of supported relations (x)**.

In an actual coupling, it is desirable that all relations which can be imported by the receiving system are then also delivered by the exporting system. Manual updating is also possible. Under no circumstances should the tables be interpreted as meaning that only the products that have the same marked relations (**x**) can be coupled!

9 Standard VDV Route Network / Timetable Interface Description

9.1 Data Model Structure

The data descriptions are divided into 6 groups, based on content:

- Calendar data
- Location data
- Operating data
- Network data
- Route data
- Timetable data

Each area is introduced by a short explanation about its basic concept.

The meaning of the relations, as well as their attributes, is explained using short descriptions. Data types and key properties for the attributes are listed in table form.

9.1.1 Notation System

- Relations which are necessary for transferring network definitions and timetable data into an AVLC system, are indicated in the "Required for" column with the letters "AVLC". “**AVLC**” (bold) indicates a key attribute, which is used for data matching with an Automatic Vehicle Location and Control System (AVLC).

- The key property of the attributes is indicated by a “P“ when it is the primary key. The keys are generally compound in nature, with the result that the record is only uniquely identifiable when all the key attributes are examined together. Attributes which enable clear record access are indicated by a 'C'.

- Attributes which were not present in the VDV Data Model v. 4.1 are indicated in the relation description using *italics*.
- Value ranges may differ (in that they are usually larger) than those in data model 4.1. No specific reference is made here.

9.1.2 Data Types

The data types used in the Network / Timetable Interface Description have been taken from the VDV Data Model v. 4.1. Here they are explained with examples:

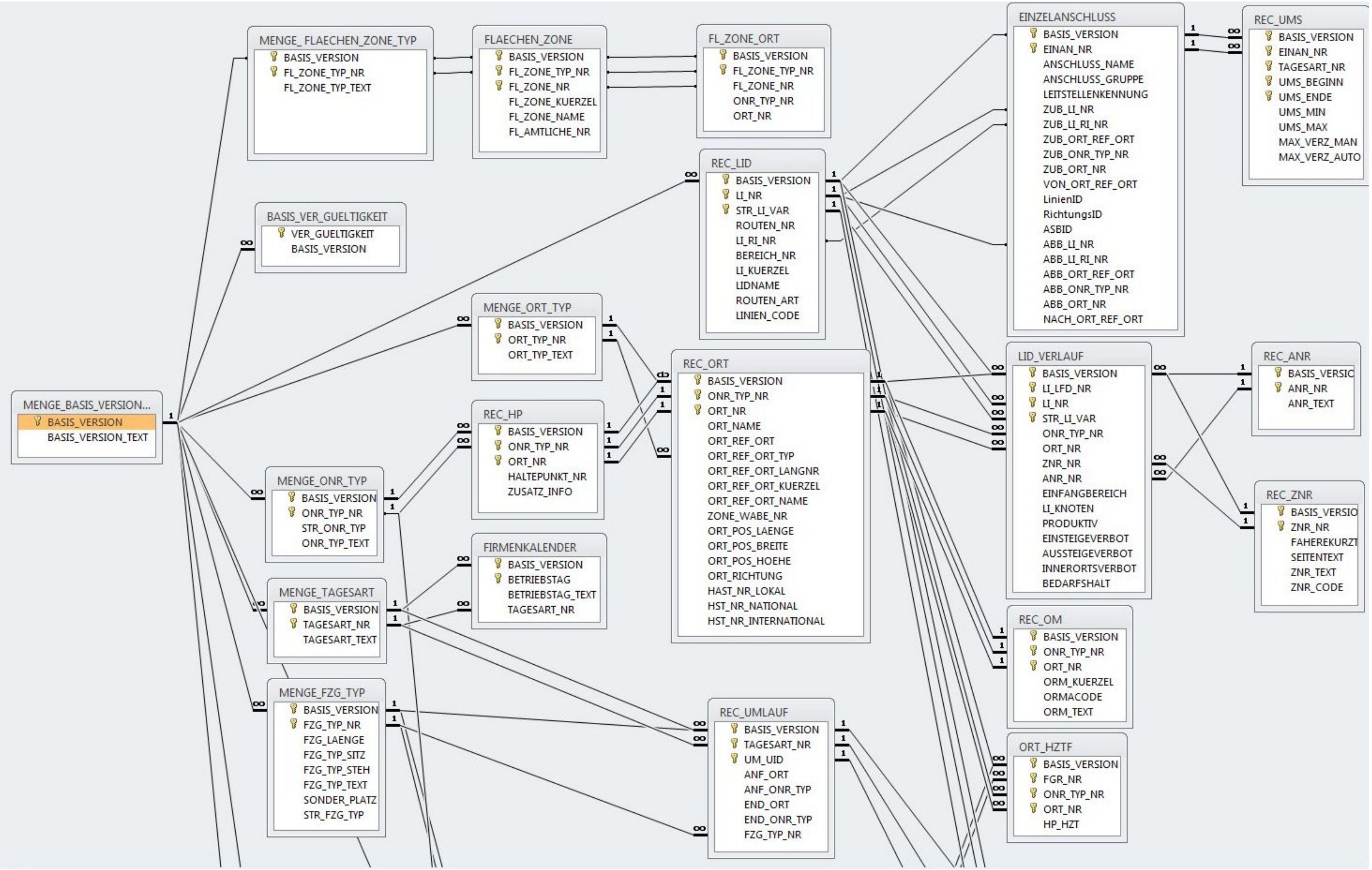
decimal (x)	Decimal value, whereby x represents the maximum number of places
char(x)	Character string, whereby x represents the maximum number of characters
boolean	Logical type : 0 = FALSE / 1 = TRUE

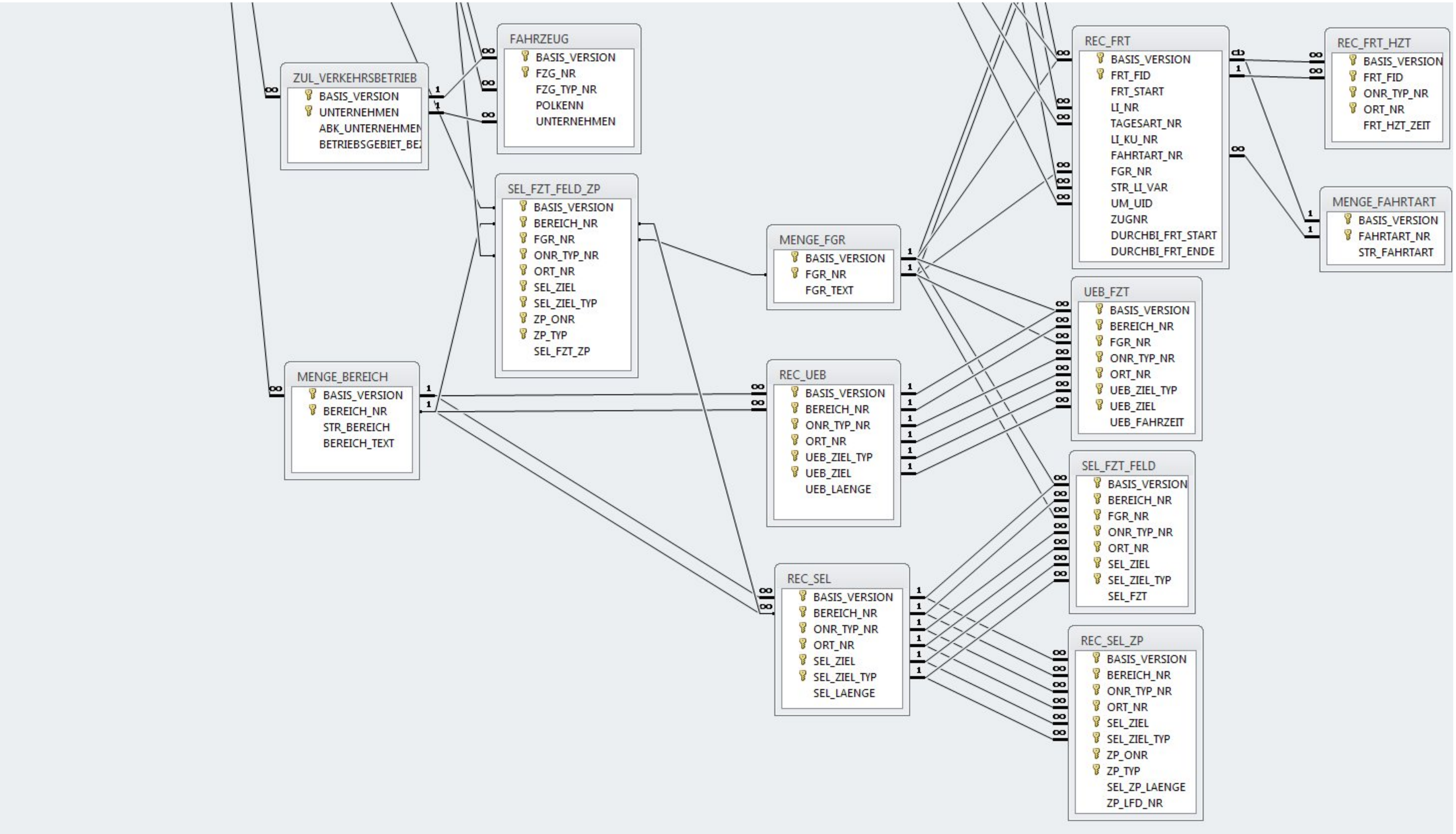
9.1.3 Value Ranges

9.1.4 Times

All times are managed in seconds.

9.1.5 Diagram of the Data Model


MENGE_FLAECHEN_ZONE_TYP
BASIS_VERSION
FL_ZONE_TYP_NR
FL_ZONE_TYP_TEXT
FLAECHEN_ZONE
BASIS_VERSION
FL_ZONE_TYP_NR
FL_ZONE_NR
FL_ZONE_KUERZEL
FL_ZONE_NAME
FL_AMTLICHE_NR
FL_ZONE_ORT
BASIS_VERSION
FL_ZONE_TYP_NR
FL_ZONE_NR
ONR_TYP_NR
ORT_NR
EINZELANSCHLUSS
BASIS_VERSION
EINAN_NR
ANSCHLUSS_NAME
ANSCHLUSS_GRUPPE
LEITSTELLENKENNUNG
ZUB_LI_NR
ZUB_LI_RI_NR
ZUB_ORT_REF_ORT
ZUB_ONR_TYP_NR
ZUB_ORT_NR
VON_ORT_REF_ORT
LinienID
RichtungsID
ASBID
ABB_LI_NR
ABB_LI_RI_NR
ABB_ORT_REF_ORT
ABB_ONR_TYP_NR
ABB_ORT_NR
NACH_ORT_REF_ORT
REC_UMS
BASIS_VERSION
EINAN_NR
TAGESART_NR
UMS_BEGINN
UMS_ENDE
UMS_MIN
UMS_MAX
MAX_VERZ_MAN
MAX_VERZ_AUTO
BASIS_VER_GUELTIGKEIT
VER_GUELTIGKEIT
BASIS_VERSION
REC_LID
BASIS_VERSION
LI_NR
STR_LI_VAR
ROUTEN_NR
LI_RI_NR
BEREICH_NR
LI_KUERZEL
LIDNAME
ROUTEN_ART
LINIEN_CODE
MENGE_ORT_TYP
BASIS_VERSION
ORT_TYP_NR
ORT_TYP_TEXT
MENGE_BASIS_VERSION...
BASIS_VERSION
BASIS_VERSION_TEXT
REC_HP
BASIS_VERSION
ONR_TYP_NR
ORT_NR
HALTEPUNKT_NR
ZUSATZ_INFO
REC_ORT
BASIS_VERSION
ONR_TYP_NR
ORT_NR
ORT_NAME
ORT_REF_ORT
ORT_REF_ORT_TYP
ORT_REF_ORT_LANGNR
ORT_REF_ORT_KUERZEL
ORT_REF_ORT_NAME
ZONE_WABE_NR
ORT_POS_LAENGE
ORT_POS_BREITE
ORT_POS_HOEHE
ORT_RICHTUNG
HAST_NR_LOKAL
HST_NR_NATIONAL
HST_NR_INTERNATIONAL
LID_VERLAUF
BASIS_VERSION
LI_LFD_NR
LI_NR
STR_LI_VAR
ONR_TYP_NR
ORT_NR
ZNR_NR
ANR_NR
EINFANGBEREICH
LI_KNOTEN
PRODUKTIV
EINSTEIGEVERBOT
AUSSTEIGEVERBOT
INNERORTSVERBOT
BEDARFSHALT
REC_ANR
BASIS_VERSIC
ANR_NR
ANR_TEXT
REC_ZNR
BASIS_VERSIO
ZNR_NR
FAHEREKURZT
SEITENTEXT
ZNR_TEXT
ZNR_CODE
MENGE_ONR_TYP
BASIS_VERSION
ONR_TYP_NR
STR_ONR_TYP
ONR_TYP_TEXT
FIRMENKALENDER
BASIS_VERSION
BETRIEBSTAG
BETRIEBSTAG_TEXT
TAGESART_NR
MENGE_TAGESART
BASIS_VERSION
TAGESART_NR
TAGESART_TEXT
REC_OM
BASIS_VERSION
ONR_TYP_NR
ORT_NR
ORM_KUERZEL
ORMACODE
ORM_TEXT
MENGE_FZG_TYP
BASIS_VERSION
FZG_TYP_NR
FZG_LAENGE
FZG_TYP_SITZ
FZG_TYP_STEH
FZG_TYP_TEXT
SONDER_PLATZ
STR_FZG_TYP
REC_UMLAUF
BASIS_VERSION
TAGESART_NR
UM_UID
ANF_ORT
ANF_ONR_TYP
END_ORT
END_ONR_TYP
FZG_TYP_NR
ORT_HZTF
BASIS_VERSION
FGR_NR
ONR_TYP_NR
ORT_NR
HP_HZT


FAHRZEUG
BASIS_VERSION
FZG_NR
FZG_TYP_NR
POLKENN
UNTERNEHMEN
ZUL_VERKEHRSBETRIEB
BASIS_VERSION
UNTERNEHMEN
ABK_UNTERNEHMEN
BETRIEBSGEBIET_BEZ
SEL_FZT_FELD_ZP
BASIS_VERSION
BEREICH_NR
FGR_NR
ONR_TYP_NR
ORT_NR
SEL_ZIEL
SEL_ZIEL_TYP
ZP_ONR
ZP_TYP
SEL_FZT_ZP
MENGE_FGR
BASIS_VERSION
FGR_NR
FGR_TEXT
REC_FRT
BASIS_VERSION
FRT_FID
FRT_START
LI_NR
TAGESART_NR
LI_KU_NR
FAHRTART_NR
FGR_NR
STR_LI_VAR
UM_UID
ZUGNR
DURCHBI_FRT_START
DURCHBI_FRT_ENDE
REC_FRT_HZT
BASIS_VERSION
FRT_FID
ONR_TYP_NR
ORT_NR
FRT_HZT_ZEIT
MENGE_FAHRTART
BASIS_VERSION
FAHRTART_NR
STR_FAHRTART
UEB_FZT
BASIS_VERSION
BEREICH_NR
FGR_NR
ONR_TYP_NR
ORT_NR
UEB_ZIEL_TYP
UEB_ZIEL
UEB_FAHRZEIT
MENGE_BEREICH
BASIS_VERSION
BEREICH_NR
STR_BEREICH
BEREICH_TEXT
REC_UEB
BASIS_VERSION
BEREICH_NR
ONR_TYP_NR
ORT_NR
UEB_ZIEL_TYP
UEB_ZIEL
UEB_LAENGE
SEL_FZT_FELD
BASIS_VERSION
BEREICH_NR
FGR_NR
ONR_TYP_NR
ORT_NR
SEL_ZIEL
SEL_ZIEL_TYP
SEL_FZT
REC_SEL
BASIS_VERSION
BEREICH_NR
ONR_TYP_NR
ORT_NR
SEL_ZIEL
SEL_ZIEL_TYP
SEL_LAENGE
REC_SEL_ZP
BASIS_VERSION
BEREICH_NR
ONR_TYP_NR
ORT_NR
SEL_ZIEL
SEL_ZIEL_TYP
ZP_ONR
ZP_TYP
SEL_ZP_LAENGE
ZP_LFD_NR

9.2 Overview of the relations

Table name German	Calendar data	Table no.	
BASIS_VER_GUELTIGKEIT	BASE_VERSION_VALID	993	The validity period of the base versions
MENGE_BASIS_VERSIONEN	BASE_VERSION	485	Version management of the master, timetable and block data
FIRMENKALENDER	PERIOD	348	Allocation of day type to calendar date
MENGE_TAGESART	DAY_TYPE	290	List of day types

Table name German	Location data	Table no.	
MENGE_ONR_TYP	POINT_TYPE	998	List of functional location types (SP, DEP, TURNAROUND, LOC_MARK)
MENGE_ORT_TYP	STOP_TYPE	997	List of grouping characteristics for locations (e.g. spatial).
REC_HP	STOP_POINT	229	Definition of the network points
REC_OM	ACTIVATION_POINT	295	Assigning location markers to locations including details of coding
REC_ORT	STOP	253	Definition of a stop or a depot

Table name German	Operating data	Table no.	
FAHRZEUG	VEHICLE	443	Description of vehicles
ZUL_VERKEHRSBETRIEB	TRANSPORT_COMPANY	992	Transport authorities
MENGE_BEREICH	OPERATING_DEPARTMENT	333	Operating branch (underground train, commuter train, etc.)
MENGE_FZG_TYP	VEHICLE_TYPE	293	Description of vehicle types
REC_ANR	ANNOUNCEMENT	996	List of announcement texts
REC_ZNR	DESTINATION	994	List of trip destinations (destination numbers)

Table name German	Network data	Table no.	
REC_SEL	LINK	299	Defined directional connections between two points in the network
REC_SEL_ZP	POINT_ON_LINK	995	Definition of the intermediate points of a path
MENGE_FGR	TIMING_GROUP	222	Definition of the written descriptions of the travel time groups

Table name German	Network data	Table no.	
ORT_HZTF	WAIT_TIME	999	Stopping times per travel time group and location
SEL_FZT_FELD	TRAVEL_TIME	282	Travel time for defined path sections
REC_UEB	DEAD_RUN	225	Defined directional connections between two points in the network for deadheads
UEB_FZT	DEAD_RUN_TIME	247	Deadhead time for defined path sections
MENGE_FAHRTART	JOURNEY_TYPE	332	List of trip types
FLAECHEN_ZONE	ZONE	571	Description of the zones
FL_ZONE_ORT	ZONE_POINT	539	Allocation of stopping points and border points to the zones.
MENGE_FLAECHEN_ ZONE_TYP	ZONE_TYPE	572	List of the types of zones.
SEL_FZT_FELD_ZP	POINT_ON_LINK_TRA VEL_TIME	540	For the defined path sections, this contains the scheduled travel time from the stopping point to the border point.

Table name German	Route data	Table no.	
LID_VERLAUF	ROUTE_SEQUENCE	246	Pattern sequence within the route
REC_LID	LINE	226	Route description

Table name German	Timetable data	Table no.	
REC_FRT	JOURNEY	715	Trip definition
REC_FRT_HZT	JOURNEY_WAIT_TIM E	308	Trip-specific waiting time at the stopping point
REC_UMLAUF	BLOCK	310	Description of the vehicle blocks

9.3 Calendar data

9.3.1 BASE_VERSION_VALID (BASIS_VER_GUELTIGKEIT) (993)

Description: Validity of the base versions. At any given point in time, the most valid version is the one which was begun most recently (expressed by the date on which it was first created, BASE_VERSION_VALID (VER_GUELTIGKEIT) attribute)

Table: BASE_VERSION_VALID (BASIS_VER_GUELTIGKEIT)

Key	Relation attributes	Data type	Value range	Required for	Description
P	BASE_VERSION_VALID (VER_GUELTIGKEIT)	decimal (8)	>0	AVLC	Date from which the general version is valid. Example: The number 19951231 means 31st December 1995
	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version

Links to other relations:

The primary key of BASE_VERSION_VALID is a secondary key in	BASE_VERSION_VALID has the following secondary key(s):
Non applicable	BASE_VERSION

9.3.2 BASE_VERSION (MENGE_BASIS_VERSIONEN) (485)

Description:	Valid versions for network, structural and timetable data. By being able to refer to a version number, it is possible to save several network and structural data versions side by side. From the BASE_VERSION_VALID (BASIS_VER_GUELTIGKEIT) table, you can tell which base version is valid on a certain day.

Table: BASE_VERSION (MENGE_BASIS_VERSIONEN)

Key	Relation attributes	Data type	Value range	Required for	Description
P	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
	BASE_VERSION_DESC (BASIS_VERSION_TEXT)	char(40)	ISO 8859-1	AVLC	Description of the general version

Links to other relations:	
The primary key of BASE_VERSION is a secondary key in	BASE_VERSION has the following secondary key(s):
All other relations of the route network / timetable interface description	Non applicable

9.3.3 PERIOD (FIRMENKALENDER) (348)

Description: Allocation of a day type to the calendar date for the operating day in question (only one day type can be assigned to each operating day)

Table: PERIOD (FIRMENKALENDER)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	OPERATING_DAY (BETRIEBSTAG)	decimal (8)	>0		Calendar date as the identifier of an operational day (may differ from the calendar day with regard to start and end times). Example: The number 19951231 means 31st December 1995
	OPERATING_DAY_DESC (BETRIEBSTAG_TEXT)	char(40)	ISO 8859-1		Description of the operational day
	DAY_TYPE_NO (TAGESART_NR)	decimal (3)	1 - 999		Identifier of the day type[1]

Links to other relations:	
The primary key of the PERIOD is a secondary key in	PERIOD has the following secondary keys:
Non applicable	BASE_VERSION

[1] It is necessary to check whether the entire range of values can be used in AVLC operation. Many transport authorities use components, whose day type numbers are restricted to the value range of 1 - 99

DAY_TYPE

9.3.4 DAY_TYPE (MENGE_TAGESART) (290)

Description: List of all types of operational days

Table: DAY_TYPE (MENGE_TAGESART)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	DAY_TYPE_NO (TAGESART_NR)	decimal (3)	1 - 999	**AVLC**	Day type label
	DAY_TYPE_DESC (TAGESART_TEXT)	char(40)	ISO 8859-1	AVLC	Description of the day type

Links to other relations:	
The primary key of DAY_TYPE is a secondary key in	DAY_TYPE has the following secondary key(s):
JOURNEY PERIOD BLOCK	BASE_VERSION

9.4 Location data

9.4.1 POINT_TYPE (MENGE_ONR_TYP) (998)

Description: List of functional location types (SP (bus stop), DEP (depot), LOC_MARK (location marker), TLP (traffic lights))

Table: POINT_TYPE (MENGE_ONR_TYP)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1, C_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 7	AVLC	Identifier of the functional type of a location 1: Stopping point 2: Depot point 3: Location marker 4: Traffic lights 5: Intermediate points 6: Depot point 7: Border point
C_2	POINT_TYPE_ABBR (STR_ONR_TYP)	char(6)	ISO 8859-1	AVLC	Abbreviation for the location type (SP, DEP, LOC_MARK, TLP, BORDERPOINT)
	POINT_TYPE_DESC (ONR_TYP_TEXT)	char(40)	ISO 8859-1	AVLC	Description of the functional type of a location

Links to other relations:	
The primary key of POINT_TYPE is a secondary key in	POINT_TYPE has the following secondary key(s):
STOP LINK DEAD_RUN DEAD_RUN_TIME STOP_POINT TRAVEL_TIME WAIT_TIME ACTIVATION_POINT POINT_ON_LINK JOURNEY_WAIT_TIME BLOCK ZONE_POINT	BASE_VERSION

9.4.2 STOP_TYPE (MENGE_ORT_TYP) (997)

Description: List of location grouping features (e.g. spatial)

Table: STOP_TYPE (MENGE_ORT_TYP)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	STOP_TYPE (ORT_TYP_NR)	decimal (2)	1 - 2	AVLC	Identifier of the location grouping feature 1: Stop 2: Depot
	STOP_TYPE_DESC (ORT_TYP_TEXT)	char(40)	ISO 8859-1	AVLC	Description of the location grouping feature

Links to other relations:	
The primary key of STOP_TYPE is a secondary key in	STOP_TYPE has the following secondary key(s):
	BASE_VERSION

9.4.3 STOP_POINT (REC_HP) (229)

Description: Points are the smallest units in timetable scheduling. Generally, passengers get on and off at a stopping point. Each stopping point must be allocated to a bus stop or depot. A bus stop / depot can have a maximum of 100 stopping points assigned to it. No stopping points with the same number are allowed for one bus stop/depot.

Table: STOP_POINT (REC_HP)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Identifier of functional type of a location
P_3	POINT_NO (ORT_NR)	decimal (6)	>0	**AVLC**	Location identifier per functional location type
	STOP_POINT_NO (HALTEPUNKT_NR)	decimal (2)	0 - 99	**AVLC**	Identifier of a stopping point within a reference location (point on the network)
	STOP_POINT_DESC (ZUSATZ_INFO)	char(40)	ISO 8859-1	AVLC	Description of the stopping point

Links to other relations:

The primary key of STOP_POINT is a secondary key in	STOP_POINT has the following secondary key(s):
Non applicable	STOP BASE_VERSION POINT_TYPE

9.4.4 ACTIVATION_POINT (REC_OM) (295)

Description:	Assigning location markers to locations including details of coding Explanation: For the purpose of tracking vehicle positions, AVLC systems can use (in addition to other possibilities) wayside location beacons, which transmit a signal when a vehicle passes or send a specific code in response to an active vehicle request. The locations of such beacons can be stored as LOCATION MARKER (ORTSMARKEN) in the ACTIVATION_POINT (REC_OM) relation. A LOCATION MARKER (ORTSMARKE) is essentially a location of a specific type, to which a specific code has been assigned, which in turn allows the system to update the vehicle position when it passes by. The LOCATION MARKER therefore has a 1:1 relationship with a STOP saved in STOP (REC_ORT).

Table: ACTIVATION_POINT (REC_OM)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1, C_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	POINT_TYPE (ONR_TYP_NR)	decimal (2)	3 - 4		Identifier of the functional type of a location
P_3	POINT_NO (ORT_NR)	decimal (6)	>0		Location identifier per functional location type <location number>
C_2	ACT_POINT_ABBR (ORM_KUERZEL)	char(6)	ISO 8859-1		Unique abbreviation
	ACT_POINT_CODE (ORMACODE)	decimal (5)	1 - 32765		Location marker coding
	ACT_POINT_DESC (ORM_TEXT)	char(40)	ISO 8859-1		Location marker description

Links to other relations:

The primary key of ACTIVATION_POINT is a secondary key in	ACTIVATION_POINT, which has the following secondary key(s):
Non applicable	BASE_VERSION POINT_TYPE

STOP

9.4.5 STOP (REC_ORT) (253)

Description: Description of locations. All the points on the network are contained in this relation. There is also a description of how the network points are formed into area groups. A bus stop / depot can be made up of several stopping points (e.g. when travelling a route in both directions). In this relation, this is highlighted by references between the network points which belong together. A bus stop / depot can have a maximum of 100 stopping points assigned to it. No stopping points with the same number are allowed for one bus stop/depot. The code (STOP_ABBR (ORT_REF_ORT_KUERZEL)) and the (STOP_NO (ORT_REF_ORT)) number must be unique across all stops and depots.

Table: STOP (REC_ORT)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 7	AVLC	Identifier of the functional type of a location
P_3	POINT_NO (ORT_NR)	decimal (6)	>0	**AVLC**	Location identifier per functional location type
	POINT_DESC (ORT_NAME)	char (40)	ISO 8859-1	AVLC	Description of the location (network point)
	STOP_NO (ORT_REF_ORT) [1)] [2)]	decimal (6)	>0[2)]	**AVLC**	Unique location number of a reference point for the purpose of (area) grouping
	STOP_TYPE (ORT_REF_ORT_TYP) [1)]	decimal (2)	1 - 2	AVLC	Location type of a given reference point for (area) grouping
	STOP_LONG_NO (ORT_REF_ORT_LANGNR) [1)]	decimal (7)	> 0, NULL	AVLC	Unique number given of a reference location within the traffic system
	STOP_ABBR (ORT_REF_ORT_KUERZEL) [1)]	char(8)	ISO 8859-1	AVLC	Unique abbreviation for a reference location

	STOP_DESC (ORT_REF_ORT_NAME) 1)	char(40)	ISO 8859-1	AVLC	Name of the reference location
	ZONE_CELL_NO (ZONE_WABE_NR) 1)	decimal (5)	>0, NULL	AVLC	Describes which zone / cell the reference location belongs to for the purpose of fare calculation.
	POINT_LONGITUDE *(ORT_POS_LAENGE)* 4)	decimal (10)	+/- 1800000000		Longitude in WGS 84 format: dddmmssnnn (degrees, minutes, seconds with decimal places, no preceding sign (+) represents e.L. (east longitude) A minus sign (-) represents w.L. (west longitude))
	POINT_LATTITUDE *(ORT_POS_BREITE)* 4)	decimal (10)	+/- 900000000		Latitude in WGS 84 format: dddmmssnnn (degrees, minutes, seconds with decimal places, no preceding sign (+) represents n.L. (north latitude) A minus sign (-) represents s.L. (south latitude)).
	POINT_ELEVATION *(ORT_POS_HOEHE)*4)	decimal (10)			WGS 84 format, application: Lift / elevator or multi-storey stop area.
	POINT_HEADING *(ORT_RICHTUNG)* 4)	decimal (3)	0 - 359		Direction of vehicle entrance into the stopping point 0 – north, 90 – east, 180 – south, 270 – west
	STOP_NO_LOCAL (HAST_NR_LOKAL)	decimal (9)	>0		Local stop number (additional number that uniquely identifies the stop) [2]
	STOP_NO_NATIONAL (HST_NR_NATIONAL)	decimal (9)	>0		National stop number (additional number that uniquely defines the stop, e.g. in Switzerland DIDOK) [3]

[2] Optional

[3] Optional

	STOP_NO_INTERNATIONAL (HST_NR_INTERNATIONAL)	char(30)			International stop ID (additional number that uniquely defines the stop, e.g. in the IFOPT /NeTEx, NAPTAN)[4]

Comments:

The attributes are only interpreted if POINT_TYPE (ONR_TYP_NR) = 1 or 2

2) IN AVLC operation, it is necessary to check whether the entire range of values can be used. Many transport authorities use equipment that only allows a range of 1 - 9999.

3) IN AVLC operation, it is necessary to check whether the entire range of values from 1 - 99,999 can be used. Many transport authorities use equipment that only allows a range of 1 - 9999 for the zone numbers.

4) These attributes are optional

Links to other relations:	
The primary key of STOP is a secondary key in	STOP has the following secondary key(s):
WAIT_TIME JOURNEY_WAIT_TIME ACTIVATION_POINT POINT_ON_LINK DEAD_RUN LINK STOP_POINT DEAD_RUN_TIME BLOCK ZONE_POINT	BASE_VERSION POINT_TYPE

[4] Optional

9.5 Operating data

9.5.1 VEHICLE (FAHRZEUG) (443)

Description: Description of vehicles

Table: Vehicle					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	VEHICLE_NO (FZG_NR)	decimal (4)	>0		Identifier of the vehicle <Vehicle number >
	VEHICLE_TYPE (FZG_TYP_NR)	decimal (3)	1-252, NULL		Identifier of vehicle type
	VEHICLE_REG (POLKENN)	char(20)	ISO 8859-1		Police registration
	COMPANY (UNTERNEHMEN)	decimal (3)	>0, NULL		Identifier of the transport company

Links to other relations:	
The primary key of VEHICLE is a secondary key in	VEHICLE has the following secondary key(s):
Non applicable	BASE_VERSION VEHICLE_TYPE TRANSPORT_COMPANY

9.5.2 TRANSPORT_COMPANY (ZUL_VERKEHRSBETRIEB) (992)

Description: List of transport companies involved in the public transport system

Table: TRANSPORT_COMPANY (ZUL_VERKEHRSBETRIEB)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	COMPANY (UNTERNEHMEN)	decimal (3)	>0		Identifier of the transport company
	COMPANY_ABBR (ABK_UNTERNEHMEN)	char(6)	ISO 8859-1		Abbreviation of the transport company
	BUSINESS_AREA_DESC (BETRIEBSGEBIET_BEZ)	char(40)	ISO 8859-1		Description of the business area (operational branch)

Links to other relations:	
The primary key of TRANSPORT_COMPANY is a secondary key in	TRANSPORT_COMPANY has the following secondary key(s):
VEHICLE	BASE_VERSION

9.5.3 OPERATING_DEPARTMENT (MENGE_BEREICH) (333)

Description: A variety of valid network areas (operating branches) exist when various modes of transport are made available (bus, city railway, underground system etc.) either on separate or on the same routes.

Table: OPERATING_DEPARTMENT (MENGE_BEREICH)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1, C_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252	**AVLC**	Identifier of the network area (operational branch)
C_2	OP_DEP_ABBR (STR_BEREICH)	char(6)	ISO 8859-1	AVLC	Abbreviation for the operational branch/department
	OP_DEP_DESC (BEREICH_TEXT)	char(40)	ISO 8859-1	AVLC	Description of the operational branch/department

Links to other relations:	
The primary key of OPERATING_DEPARTMENT is a secondary key in	OPERATING_DEPARTMENT has the following secondary key(s):
LINK DEAD_RUN	BASE_VERSION

9.5.4 VEHICLE_TYPE (MENGE_FZG_TYP) (293)

Description: Description of the vehicle types

Table: VEHICLE_TYPE (MENGE_FZG_TYP)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	VH_TYPE_NO (FZG_TYP_NR)	decimal (3)	1 - 252	**AVLC**	Identifier of the vehicle type
	VH_TYPE_LENGTH (FZG_LAENGE)	decimal (2)	>0	AVLC	Total length of vehicle (in metres)
	VH_TYPE_SEAT (FZG_TYP_SITZ)	decimal (3)	>=0	AVLC	Vehicle seating capacity
	VH_TYPE_STAND (FZG_TYP_STEH)	decimal (3)	>=0	AVLC	Vehicle standing capacity
	VH_TYPE_DESC (FZG_TYP_TEXT)	char(40)	ISO 8859-1	AVLC	Description of vehicle type
	VH_TYPE_SPEC_SEAT (SONDER_PLATZ)	decimal (3)	>=0	AVLC	Number of special seats (suitable for disabled passengers) in the vehicle
	VH_TYPE_ABBR (STR_FZG_TYP)	char(6)	ISO 8859-1	AVLC	Abbreviation of vehicle type

Links to other relations:	
The primary key of VEHICLE_TYPE is a secondary key in	VEHICLE_TYPE has the following secondary key(s):
VEHICLE BLOCK	BASE_VERSION

9.5.5 ANNOUNCEMENT (REC_ANR) (996)

Description: List of vehicle announcement texts (there was previously no such relation in VDV Data Model 4.1)

Table: ANNOUNCEMENT (REC_ANR)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	*ANN_NO (ANR_NR)*	decimal (4)	1 - 9999		Announcement text number
	ANN_DESC (ANR_TEXT)	char(200)	ISO 8859-1		Announcement text

Links to other relations:	
The primary key of ANNOUNCEMENT is a secondary key in	ANNOUNCEMENT has the following secondary key(s):
ROUTE_SEQUENCE	BASE_VERSION

9.5.6 DESTINATION (REC_ZNR) (994)

Description: List of journey destinations displayed on the vehicle

Table: DESTINATION (REC_ZNR)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	DEST_NO (ZNR_NR)	decimal (4)	0 - 9999		Identifier of the destination display <destination number>. DEST_NO (ZNR_NR) 0 is used to delete the display.
	DEST_BRIEF_TEXT (FAHRERKURZTEXT)	char(44)	ISO 8859-1		Brief destination display text
	DEST_SIDE_TEXT (SEITENTEXT)	char(160)	ISO 8859-1		Text for the side destination display
	DEST_FRONT_TEXT (ZNR_TEXT)	char(160)	ISO 8859-1		Text for the front destination display
	DEST_CODE (ZNR_CODE)	char(68)	ISO 8859-1		Control code for destination text displays

Links to other relations:	
The primary key of DESTINATION is a secondary key in	DESTINATION has the following secondary key(s):
ROUTE_SEQUENCE	BASE_VERSION

9.6 Network data

9.6.1 LINK (REC_SEL) (299)

Description: Defines directed (one-way) connections in the network by indicating the geometric locations (bus stops / stopping points or depots / depot points), which in turn form the beginning and end of a route. This means that routes in two different directions can exist between two stopping points. The connection distance is given in metres.

Table: LINK (REC_SEL)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252	AVLC	Identifier of the operational branch
P_3	FROM_POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Identifier of the functional type of the route starting point
P_4	FROM_POINT_NO (ORT_NR)	decimal (6)	>0	AVLC	
P_6	TO_POINT_NO (SEL_ZIEL)	Decimal (6)	>0	AVLC	Location number of the route end point
P_5	TO_POINT_TYPE (SEL_ZIEL_TYP)	decimal (2)	1 - 2	AVLC	Location type of the route end point
	LINK_DISTANCE (SEL_LAENGE)	decimal (5)	1 - 81890	AVLC	Length of route (junction-oriented), in metres

Links to other relations:	
The primary key of LINK is a secondary key in	LINK has the following secondary key(s):
TRAVEL_TIME	BASE_VERSION OPERATING_DEPARTMENT POINT_TYPE STOP

9.6.2 POINT_ON_LINK (REC_SEL_ZP) (995)

Description: Definition of intermediate points (location markers, traffic lights, intermediate points) on a route. With the help of intermediate points it is possible to define the graphical display of a route between two stopping points. The POINT_ON_LINK_SERIAL_NO *(ZP_LFD_NR)* attribute defines the order of the intermediate points on the route.

Table: POINT_ON_LINK (REC_SEL_ZP)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252		Identifier of the operational branch
P_3	FROM_POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2		Location type of the starting point of the route.
P_4	FROM_POINT_NO (ORT_NR)	decimal (6)	>0		Location number of the starting point of the route.
P_6	TO_POINT_NO (SEL_ZIEL)	decimal (6)	>0		Location number of the end point of the route.
P_5	TO_POINT_TYPE (SEL_ZIEL_TYP)	decimal (2)	1 - 2		Location type of the end point of the route.
P_8	POINT_TO_LINK_NO (ZP_ONR)	decimal (6)	>0		Location number of an intermediate point on the route (junction-oriented)
P_7	POINT_TO_LINK_TYPE (ZP_TYP)	decimal (2)	3 - 7		Location type of an intermediate point or a border point on the route (junction-oriented)
	POINT_TO_DISTANCE (SEL_ZP_LAENGE)	decimal (5)	1-81890, NULL		Length of route between the beginning and ending point in metres

	POINT_ON_LINK_SERIAL_NO (ZP_LFD_NR)	decimal (3)	>0, NULL		Serial number of the intermediate point counted from the beginning of the route

Links to other relations:	
Primary key of POINT_ON_LINK is a secondary key in	POINT_ON_LINK has the following secondary key(s):
Non applicable	BASE_VERSION LINK POINT_TYPE STOP

9.6.3 TIMING_GROUP (MENGE_FGR) (222)

Description: Contains the text description for the travel time groups. The number of the travel time group indicates a day-time interval during which the travel or stopping times are valid.

Table: TIMING_GROUP (MENGE_FGR)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	TIMING_GROUP_NO (FGR_NR)	decimal (9)	>0	**AVLC**	Identifier of the travel time group[5]
	TIMING_GROUP_DESC (FGR_TEXT)	char(40)	ISO 8859-1	AVLC	Description of the travel time group

Links to other relations:	
The primary key of TIMING_GROUP is a secondary key in	TIMING_GROUP has the following secondary key(s):
WAIT_TIME TRAVEL_TIME DEAD_RUN_TIME	BASE_VERSION

[5] It is necessary to check whether the entire range of values can be used in AVLC operation. Many transport authorities use components whose stop numbers are restricted to the value range of 1 - 65535

9.6.4 WAIT_TIME (ORT_HZTF) (999)

Description: Stopping times per travel time group and location

Table: WAIT_TIME (ORT_HZTF)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	TIMING_GROUP_NO (FGR_NR)	decimal (9)	>0	AVLC	Identifier of the travel time group
P_3	POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Identifier of the functional type of a location <location type>
P_4	POINT_NO (ORT_NR)	decimal (6)	>0	AVLC	Location identifier per functional location type <location number>
	WAIT_TIME (HP_HZT)	decimal (6)	0-65532	AVLC	Stopping time at a location per travel time group

Links to other relations:	
The primary key of WAIT_TIME is a secondary key in	WAIT_TIME has the following secondary key(s)
Non applicable	BASE_VERSION POINT_TYPE TIMING_GROUP STOP

9.6.5 TRAVEL_TIME (SEL_FZT_FELD) (282)

Description: Contains the scheduled travel time for the defined route sections. The time needed to cover the route can depend on the time of day. Therefore, a number of travel times could apply to the same stretch. The various travel times are uniquely identified by a travel time group. The travel times are given in seconds.

Table: TRAVEL_TIME (SEL_FZT_FELD)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252	AVLC	Identifier of the operational branch
P_3	TIMING_GROUP_NO (FGR_NR)	decimal (9)	>0	AVLC	Identifier of the travel time group
P_4	FROM_POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Location type of the starting point of the route.
P_5	FROM_POINT_NO (ORT_NR)	decimal (6)	>0	AVLC	Location number of the starting point of the route.
P_7	TO_POINT_NO (SEL_ZIEL)	decimal (6)	>0	AVLC	Location number of the end point of the route.
P_6	TO_POINT_TYPE (SEL_ZIEL_TYP)	decimal (2)	1 - 2	AVLC	Location type of the end point of the route.
	TRAVEL_TIME (SEL_FZT)	decimal (6)	0-65532	AVLC	Section travel time per travel time group (junction-oriented) in seconds

Links to other relations:	
The primary key of TRAVEL_TIME is a secondary key in	TRAVEL_TIME has the following secondary key(s):
Non applicable	BASE_VERSION LINK TIMING_GROUP OPERATING_DEPARTMENT POINT_TYPE STOP

9.6.6 DEAD_RUN (REC_UEB) (225)

Description: Defines directed (one-way) connections in the network by indicating the geometric locations (bus stops / stopping points) which form the beginning and end of the route. The DEAD_RUN (REC_UEB) relation is needed for deadheads (depot departures, depot entries, approach). Deadheads only ever consist of one connection between two points, whereby these must not be identical!

Table: DEAD_RUN (REC_UEB)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252	AVLC	Identifier of the operational branch
P_3	FROM_POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Location type of the starting point of the deadhead path.
P_4	FROM_POINT_NO (ORT_NR)	decimal (6)	>0	AVLC	Location number of the starting point of the deadhead path.
P_5	TO_POINT_TYPE (UEB_ZIEL_TYP)	decimal (2)	1 - 2	AVLC	Location type of the end point of the deadhead path
P_6	TO_POINT_NO (UEB_ZIEL)	decimal (6)	>0	AVLC	Location number of the end point of the deadhead path.
	DEAD_RUN_DISTANCE (UEB_LAENGE)	decimal (6)	1 - 81890	AVLC	Length of the deadhead path in metres.
Links to other relations:					
The primary key of DEAD_RUN is a secondary key in			DEAD_RUN has the following secondary key(s):		

DEAD_RUN_TIME	BASE_VERSION OPERATING_DEPARTMENT POINT_TYPE STOP

9.6.7 DEAD_RUN_TIME (UEB_FZT) (247)

Description: Travel time of the deadhead. Contains the scheduled travel time for the defined route sections. The time needed to cover the route can depend on the time of day. Therefore, a number of different travel times can apply to the same stretch. The various travel times are uniquely identified by a travel time group. The travel time of a deadhead must be greater than zero and the points (beginning / end) should not be identical!

Table: DEAD_RUN_TIME (UEB_FZT)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252	AVLC	Identifier of the operational branch
P_3	TIMING_GROUP_NO (FGR_NR)	decimal (9)	> 0	AVLC	Identifier of the travel time group
P_4	FROM_POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Location type of the starting point on the deadhead path
P_5	FROM_POINT_NO (ORT_NR)	decimal (6)	>0	AVLC	Location from-point number on the deadhead path
P_6	TO_POINT_TYPE (UEB_ZIEL_TYP)	decimal (2)	1 - 2	AVLC	Location type of the end point of the deadhead path
P_7	TO_POINT_NO (UEB_ZIEL)	decimal (6)	>0	AVLC	Location number of the end point of the deadhead path

	TRAVEL_TIME (UEB_FAHRZEIT)	decimal (6)	1-65532	AVLC	Travel time of the deadhead per travel time group, in seconds.

Links to other relations:	
The primary key of DEAD_RUN_TIME is a secondary key in	DEAD_RUN_TIME has the following secondary key(s):
Non applicable	BASE_VERSION OPERATING_DEPARTMENT TIMING_GROUP DEAD_RUN

9.6.8 JOURNEY_TYPE (MENGE_FAHRTART) (332)

Description: List of journey types

Table: JOURNEY_TYPE (MENGE_FAHRTART)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1, C_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	JOURNEY_TYPE_NO (FAHRTART_NR)	decimal (2)	1 - 4	AVLC	Identifier of the type of journey 1: Normal trip 2: Depot exit 3: Depot entry 4: Approach
C_2	JOURNEY_TYPE_DESC (STR_FAHRTART)	char(6)	ISO 8859-1	AVLC	Journey type abbreviation

Links to other relations:	
The primary key of JOURNEY_TYPE is a secondary key in	JOURNEY_TYPE has the following secondary key(s):
JOURNEY	BASE_VERSION

Note: The "approach" is a path which is used especially for route change-over journeys and empty runs.

9.6.9 ZONE (FLAECHEN_ZONE) (571)

Description: Description of the zones. A zone is a spatially connected geographical region. The individual zones in the ZONE (FLAECHEN_ZONE) table need not necessarily be disjointed. The zones can overlap.

One application of the zones is the division into political regions, the so-called administrative units.

An administrative unit is a legal body that has the territorial jurisdiction for a specific area of the county. The administrative units can be assigned on different levels. Examples include the communal administrative units of rural areas, cities, districts and municipalities.

Table: ZONE (FLAECHEN_ZONE)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1 C_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2 C_2	ZONE_TYPE_NO (FL_ZONE_TYP_NR)	decimal (1)	1 - 9		Type of the zone
P_3	ZONE_NO (FL_ZONE_NR)	decimal (6)	>0		Number of the zone / administrative unit
C_3	ZONE_ABBR (FL_ZONE_KUERZEL)	char(8)	ISO 8859-1		Abbreviation of the zone / administrative unit
	ZONE_DESC (FL_ZONE_NAME)	char(40)	ISO 8859-1		Description of the zone / administrative unit
	ZONE_ADMINISTRATIVE_NO (FL_AMTLICHE_NR)	char(20)	ISO 8859-1		Usually contains the county code.

Links to other relations:	
The primary key of ZONE is a secondary key in	ZONE has the following secondary key(s):
ZONE_POINT	BASE_VERSION ZONE_TYPE

9.6.10 ZONE_POINT (FL_ZONE_ORT) (539)

Description:	Allocation of the various location points (stopping points, depot points and area boundaries) to the zones. A single location point can be assigned to several zones. The area boundaries are assigned to the adjoining zones.

Table: ZONE_POINT (FL_ZONE_ORT)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	ZONE_TYPE_NO (FL_ZONE_TYP_NR)	decimal (1)	1 - 9		Type of the zone
P_3	ZONE_NO (FL_ZONE_NR)	decimal (6)	>0		Number of the zone / administrative unit
P_4	POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 7		Identifier of the functional type of a location <location type>
P_5	POINT_NO (ORT_NR)	decimal (6)	>0		Location identifier per functional location type

Links to other relations:	
The primary key of ZONE_POINT is a secondary key in	ZONE_POINT has the following secondary key(s):
Not relevant	BASE_VERSION POINT_TYPE STOP ZONE ZONE_TYPE

9.6.11 ZONE_TYPE (MENGE_ FLAECHEN_ZONE_TYP) (572)

Description: List of the types of zones (administrative unit)

Table: ZONE_TYPE (MENGE_ FLAECHEN_ZONE_TYP)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	ZONE_TYPE_NO (FL_ZONE_TYP_NR)	decimal (1)	1 - 9		Describes the functional type of a zone 1: Administrative unit 2: Traffic zone …
	ZONE_TYPE_DESC (FL_ZONE_TYP_TEXT)	char(40)	ISO 8859-1		Description of the type of zone

Links to other relations:	
The primary key of the ZONE_TYPE is a secondary key in	ZONE_TYPE has the following secondary keys
ZONE ZONE_POINT	BASE_VERSION

9.6.12 POINT_ON_LINK_TRAVEL_TIME (SEL_FZT_FELD_ZP) (540)

Description: For the defined path sections, this contains the scheduled travel time from the stopping point to the border point. The time needed to cover the route can depend on the travel time group. This means that a number of travel times could apply to the same stretch. The travel times are given in seconds.

Table: POINT_ON_LINK_TRAVEL_TIME (SEL_FZT_FELD_ZP)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252	AVLC	Identifier of the operational branch
P_3	TIMING_GROUP_NO (FGR_NR)	decimal (9)	>0	AVLC	Identifier of the travel time group
P_4	FROM_POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Location type of the starting point of the route.
P_5	FROM_POINT_NO (ORT_NR)	decimal (6)	>0	AVLC	Location number of the starting point of the route.
P_7	TO_POINT_NO (SEL_ZIEL)	decimal (6)	>0	AVLC	Location number of the end point of the route.
P_6	TO_POINT_TYPE (SEL_ZIEL_TYP)	decimal (2)	1 - 2	AVLC	Location type of the end point of the route.
P_8	POINT_TO_LINK_NO (ZP_ONR)	decimal (6)	>0		Location number of an intermediate point on the route (junction-oriented)
P_7	POINT_TO_LINK_TYPE (ZP_TYP)	decimal (2)	7		Location type of an intermediate point or a border point on the route (junction-oriented)

	TRAVEL_TIME (SEL_FZT_ZP)	decimal (6)	0-65532	AVLC	Section travel time per travel time group (junction-oriented) from the stopping point to the border point in seconds.

Links to other relations:	
The primary key of TRAVEL_TIME is a secondary key in	TRAVEL_TIME has the following secondary key(s):
Non applicable	BASE_VERSION LINK TIMING_GROUP OPERATING_DEPARTMENT POINT_TYPE STOP

9.7 Route data

9.7.1 ROUTE_SEQUENCE (LID_VERLAUF) (246)

Description: Describes the route by listing the bus stops / points which are stopped at in numbered sequence. Bus and stopping points (depots / depot points) may only be stopped at (served) once within the pattern sequence (no circular routes). Circular routes are reproduced by representing the repeatedly served stops as different locations (stopping points) with the same reference location (stop). The total travel time for a route cannot be zero. The same applies to distance. The beginning and end points of a route must be junctions (time-relevant locations). SEQUENCE_NO (LI_LFD_NR) describes the pattern sequence, which lists the stops and stopping points to be served as a sequence of incrementally increasing numbers (e.g. 1,4,6,8 represents a valid sequence).

Table: ROUTE_SEQUENCE (LID_VERLAUF)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1, C_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_4	SEQUENCE_NO (LI_LFD_NR)	decimal (3)	>0	AVLC	Consecutive number of the point on the route pattern
P_2, C_2	LINE_NO (LI_NR)	decimal (6)	1 - 99999	**AVLC**	Identifier of the transport supply in terms of route or direction[6]
P_3, C_3	ROUTE_ABBR (STR_LI_VAR)	char(6)	ISO 8859-1	**AVLC**	Identifier of the variant on the route (or pattern sequence in a specific direction)

[6] It is necessary to check whether the entire range of values can be used in AVLC operation. Many transport authorities use components whose route numbers are restricted to the value range of 1 - 999

C_4	POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2	AVLC	Identifier of the functional type of a location <location type>
C_5	POINT_NO (ORT_NR)	decimal (6)	>0	AVLC	Location identifier per functional location type <location number>
	DEST_NO (ZNR_NR)	decimal (4)	0-9999 (0)	AVLC	Identifier of the destination display
	ANN_NO (ANR_NR)	decimal (4)	1-9999, NULL	AVLC	Identifier of the announcement
	LOCKIN_RANGE (EINFANGBEREICH)	decimal (3)	0-256 (0)	AVLC	Area in metres within which the on-board computer recognises the bus stop / stopping point
	LINE_NODE (LI_KNOTEN)	boolean	0 - 1 (1)	AVLC	0: Not a rime-relevant location 1: Time relevant location
	PRODUCTIVE (PRODUKTIV)	boolean	0 - 1 (1)		Specifies whether it is a normal trip (with passengers) or a depot trip (e.g. a depot exit). [7]
	NO_BOARDING (EINSTEIGEVERBOT)	boolean	0 - 1 (0)		Passengers are not permitted to board the vehicle here [8] [9]

[7] The field is optional: If not delivered, the specification in TRIP_TYPE_NO (FAHRTART_NR) is valid (e.g. normal trip). Partial sections (i.e. edges defined by two table records) with PRODUCTIVE (PRODUKTIV)=0 can only exist at the start or end of a path. There must only be one connected area with PRODUCTIVE (PRODUKTIV)=1. Paths can be composed exclusively of sections with either PRODUCTIVE (PRODUKTIV)=1 or PRODUCTIVE=0.
When transferring the schedule times, both the productive and unproductive sections must be provided with times.

[8] Optional

[9] For a pass-by, both the NO_BOARDING and NO_ALIGHTING elements should be set to 1 (true).

	NO_ALIGHTING (AUSSTEIGEVERBOT)	boolean	0 - 1 (0)		Passengers are not permitted to leave the vehicle here [10] [11]
	CITY_BAN (INNERORTSVERBOT)	boolean	0 - 1 (0)		It is forbidden for passengers to board or alight vehicles within the city limits (generally applies to regional journeys from and to the countryside)[12]
	REQUEST_STOP (BEDARFSHALT)[13]	boolean	0 - 1 (0)		Bookable stop along a trip (the vehicle stops at the request of the passenger).

Links to other relations:	
The primary key of ROUTE_SEQUENCE is a secondary key in	ROUTE_SEQUENCE has the following secondary key(s):
Non applicable	BASE_VERSION LINE ANNOUNCEMENT DESTINATION STOP POINT_TYPE

[10] Optional

[11] For a pass-by, both the NO_BOARDING and NO_ALIGHTING elements should be set to 1 (true).

[12] Optional

[13] Trips with flexible usage can contain, in addition to the regular stops, so-called request stops, which can already be labelled as such in the planning programs.

9.7.2 LINE (REC_LID) (226)

Description: Allocation of the route (sometimes referred to as line) to the operational branch. The route number within a network is unique. The pattern number must be uniquely assigned to a route and pattern sequence. LINE_ABBR (LI_KUERZEL) must have the same value for all patterns on the same route (LINE_NO (LI_NR)).

Table: LINE (REC_LID)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1, C_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2, C_2	LINE_NO (LI_NR)	decimal (6)	1 - 9999	**AVLC**	Identifier of the transport supply in terms of route or direction
P_3	ROUTE_ABBR (STR_LI_VAR)	char(6)	ISO 8859-1	**AVLC**	Identifier of the variant on the route (or pattern sequence in a specific direction)
C_3	*ROUTE_NO (ROUTEN_NR)*	decimal (3)	1 - 999	**AVLC**	Unique identification of a pattern sequence in accordance with a specific route for the vehicle on-board computer
	DIRECTION (LI_RI_NR)	decimal (3)	1 - 2	AVLC	Identifier of the route direction
	OP_DEP_NO (BEREICH_NR)	decimal (3)	0 - 252	AVLC	Identifier of the operational branch
	LINE_ABBR (LI_KUERZEL)	char(6)	ISO 8859-1	AVLC	Name of the route
	LINE_DESC (LIDNAME)	char(40)	ISO 8859-1	AVLC	Description of the route
	ROUTE_TYPE (ROUTEN_ART)	decimal (2)	1 - 4	AVLC	1: Normal profile 2: Depot entry 3: Depot exit 4: Approach
	LINE_CODE (LINIEN_CODE)	decimal (2)	>0, NULL	AVLC	Identifier of the mask number for the on-vehicle display

1) 2)	ExternalLineRef (LinienID)	char(128)			Country wide unique and valid line ID

1) TeillinienID (TLID) described in VDV433 may be transmitted as 'Betreiber' (Vgl. auch VDV453/454 V3) and coded in FRT_FID.

2)

VDV433	DLID
VDV452 englisch	LINE.ExternalLineRef
VDV453	DatenAbrufenAntwort/AZBNachricht/AZBFahrplan/LinienID
SIRI-SM	StopMonitoringDelivery/MonitoredStopVisit/MonitoredVehicleJourney/LineRef
VDV454	DatenAbrufenAntwort/AUSNachricht/IstFahrt/LinienID
SIRI-PT	ProductionTimetableDelivery/DatedTimetableVersionFrame/DatedVehicleJourney/ExternalLineRef
VDV462	ServiceFrame/lines/Line/ExternalLineRef

Links to other relations:	
The primary key of LINE is a secondary key in	LINE has the following secondary key(s):
ROUTE_SEQUENCE	BASE_VERSION OPERATING_DEPARTMENT

9.8 Timetable data

9.8.1 JOURNEY (REC_FRT) (715)

Description: Journey definition in "Information on Scheduling Journeys". Result of the journey relationship investigation, according to which linked routes are brought together to form complete journey relations, also taking into account admissible route changes (reassignments). The run number is used to uniquely allocate the vehicles on a route to a timetable. In so doing, the run identify all the vehicles which are being used at a certain point in time. The run number gives no information about the number of vehicles which are being used at any given time point. The run number is unique within the route and for the time during which the vehicle in question is on the route.

Table: JOURNEY (REC_FRT)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1, C_{11}, C_{21}	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	JOURNEY_NO (FRT_FID)	decimal (10)	>0	AVLC	Identifier of journey
C_{15}, C_{24}	DEPARTURE_TIME (FRT_START)	decimal (6)	0 - 129600	**AVLC**	Journey departure time in seconds from 0:00
C_{13}	LINE_NO (LI_NR)	decimal (6)	1 - 9999	**AVLC**	Identifier of the transport supply in terms of route or direction
C_{12}, C_{22}	DAY_TYPE_NO (TAGESART_NR)	decimal (3)	1 - 999	**AVLC**	Day type label
C_{14}	RUN (LI_KU_NR)[1)]	decimal (6)	1-99, NULL	**AVLC**	Run number of a block within a route
	JOURNEY_TYPE (FAHRTART_NR)	decimal (2)	1 - 4	AVLC	Identifier of the type of journey
	TIMING_GROUP_NO (FGR_NR)	decimal (9)	>0	AVLC	Identifier of the travel time group

	ROUTE_ABBR (STR_LI_VAR)	char(6)	ISO 8859-1	AVLC	Identifier of the variant on the route (or pattern sequence in a specific direction)
C_{23}	BLOCK_NO (UM_UID)	decimal (8)	>0, NULL	**AVLC**	Identifier of the vehicle block
	TRAIN_NO (ZUGNR)	decimal (7)	>0		This is the train number that is presented to the passenger via the printed timetable.
	THROUGH_START (DURCHBI_FRT_START) [2)]	boolean	0 - 1 (0)		The field content is 1 (logic true), if passengers are allowed in the vehicle from the start of the trip (occupancy at the start of the trip > 0).
	THROUGH_END (DURCHBI_FRT_ENDE) [2)]	boolean	0 - 1 (0)		The field content is 1 (logic true), if passengers are allowed in the vehicle at the end of the trip (occupancy at the end of the trip > 0).
3)	ExternalVehicleJourneyRef (FahrtBezeichner)	char(128)			Country wide for operating day unique ID for the journey

1) With AVLC operation it is necessary to clarify whether the run numbers are actually required in the system. If yes, the value range is 1 - 99. If the transport authority and all system components work exclusively with the block number however, the run number should not be entered.

2) This is required in order to supply the balance matching processes in every AFZ system with the information concerning whether the passenger occupancy at the start or end of a trip is permitted to be > 0. It is not concerned with the actual number of occupants. This cannot be 'planned' in advance. The system only needs to know whether or not passengers can remain seated in the vehicle at the end of the trip or already be on board at the start of the trip.

These situations normally occur on circular routes or after route reassignments. This is also common on school bus journeys, as the students need to travel on several trips before reaching their destination. There is basically no defined 'trip end' at which there is zero occupancy (with the exception of drivers and on-board staff).

A through-running service can also be used where a normal trip (trip type 1) leads into a depot trip (trip type 2 and higher) or when a depot trip (trip type 2 and higher) leads into a normal trip (trip type 1).
The through-service is then valid from or to the next normal trip respectively.
This covers the situation in which passengers on a shunting trip (or relocation trip) within a stop may stay on board, even though this trip is represented as a depot trip within the system.

3)	VDV433	DFID
	VDV452 englisch	JOURNEY.ExternalVehicleJourneyRef
	VDV453	DatenAbrufenAntwort/AZBNachricht/AZBFahrplan/FahrtID/FahrtBezeichner
	SIRI-SM	StopMonitoringDelivery/MonitoredStopVisit/MonitoredVehicleJourney/FramedVehicleJourneyRef/DatedVehicleJourneyRef
	VDV454	DatenAbrufenAntwort/AUSNachricht/IstFahrt/FahrtRef/FahrtID/FahrtBezeichner
	SIRI-PT	ProductionTimetableDelivery/DatedTimetableVersionFrame/DatedVehicleJourney/FramedVehicleJourneyRef/DatedVehicleJourneyRef
	VDV462	TimetableFrame/vehicleJourneys/ServiceJourney/ExternalVehicleJourneyRef

Links to other relations:	
The primary key of JOURNEY is a secondary key in	JOURNEY has the following secondary key(s):
JOURNEY_WAIT_TIME	BASE_VERSION
	LINE
	DAY_TYPE
	TIMING_GROUP
	JOURNEY_TYPE
	BLOCK

Note on scheduling vehicle blocks: There are basically two ways of scheduling vehicle blocks from the various relations.

1. All the vehicle blocks, including deadheads, are fitted into the JOURNEY (REC_FRT) relation. The DEAD_RUN (REC_UEB) and DEAD_RUN_TIME (UEB_FZT) relations are not used. The advantage lies in the fact that a JOURNEY_NO (FRT_FID) and the valid travel time group exist for each deadhead as well as for the other journeys in this relation.
2. All the journeys, apart from the deadheads, are stored in JOURNEY (REC_FRT). If, in the course of a vehicle block, it is discovered in JOURNEY (REC_FRT) that the location number of the destination of the x-th journey does not agree with the location number of the beginning of the x+1-th journey, then a suitable deadhead has to be sought in the DEAD_RUN (REC_UEB) table. The valid travel time group for the deadhead is taken from or corresponds to that of the x-th journey. If the deadhead has no predecessor (x-th journey is missing, e.g. when departing from the depot), then the travel time group is taken from the x+1-th journey.

Note on 'missing' vehicle blocks: see 9.8.3

Note: With AVLC operation it is necessary to clarify whether deadheads can be used to create turning areas.

9.8.2 JOURNEY_WAIT_TIME (REC_FRT_HZT) (308)

Description: Journey waiting time at the bus stop. The waiting time is made up of the time it takes for passengers to board and alight including any waiting time (e.g. arriving in time to ensure the connection).

Table: JOURNEY_WAIT_TIME (REC_FRT_HZT)					
Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0		Label of the general version
P_2	JOURNEY_NO (FRT_FID)	decimal (10)	>0		Identifier of the journey
P_3	POINT_TYPE (ONR_TYP_NR)	decimal (2)	1 - 2		Location type
P_4	POINT_NO (ORT_NR)	decimal (6)	>0		Location number of stopping point
	JOURNEY_WAIT_TIME (FRT_HZT_ZEIT)	decimal (6)	**0-65532**		Journey stopping time at a bus stop (in seconds)

Links to other relations:	
The primary key of JOURNEY_WAIT_TIME is a secondary key in	JOURNEY_WAIT_TIME has the following secondary key(s):
Non applicable	BASE_VERSION POINT_TYPE JOURNEY STOP

Note: Trip-specific waiting times can only be used at the intermediate stops of a pattern. The start and terminal stops cannot be assigned trip-specific waiting times.

9.8.3 BLOCK (REC_UMLAUF) (310)

Description: Description of the vehicle blocks. Each vehicle block must begin with departure from the depot and end with access to the depot.

Table: BLOCK (REC_UMLAUF)

Key	Relation attributes	Data type	Value range	Required for	Description
P_1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P_2	DAY_TYPE_NO (TAGESART_NR)	decimal (3)	1 - 999	**AVLC**	Day type label
P_3	BLOCK_NO (UM_UID)	decimal (8)	>0	**AVLC**	Identifier of the vehicle block
	FROM_POINT_NO (ANF_ORT)	decimal (6)	>0	AVLC	Location number of the start location of a block.
	FROM_POINT_TYPE (ANF_ONR_TYP)	decimal (2)	1 - 2	AVLC	Location type of the start location of a block (type: depot)
	TO_POINT_NO (END_ORT)	decimal (6)	>0	AVLC	Location number of the end location of a block
	TO_POINT_TYPE (END_ONR_TYP)	decimal (2)	1 - 2	AVLC	Location type of the end location of a block (type: depot)
	VH_TYPE_NO (FZG_TYP_NR)	decimal (3)	1-252, NULL	AVLC	Identifier of vehicle type

Links to other relations:

The primary key of BLOCK is a secondary key in	BLOCK has the following secondary key(s):
JOURNEY	BASE_VERSION
	DAY_TYPE
	VEHICLE_TYPE

Note:
For certain import systems, information on vehicle blocks is not necessary (e.g. passenger counting, counting of disabled passengers and timetable information). That is why in some transport companies, no block scheduling is carried out.
In such a case, the exporting system assigns a "0" to BLK_ID (UM_ID) in the interface file (NULL in the database). Therefore, the block table (9.8.3) becomes an optional table, except when updating an AVLC.

10 Interface Description: Transfer Data for AVLC

The **JOURNEY CONNECTION (EINZELANSCHLUSS)** and **INTERCHANGE (REC_UMS)** tables described in this chapter allow the transfer of connection definitions and their validities, for example from a planning system to an AVLC. This ensures the AVLC has the necessary information to monitor and protect transfers. Transfer protection of third-party vehicles is also supported by the ConnectionLinkRef (ASBID), LINE_ID (LinienID) and DIRECTION_ID (RichtungsID) attributes in accordance with VDV Recommendation 453 ("Real-time Data Interface").

10.1 JOURNEY CONNECTION (EINZELANSCHLUSS) (432)

Description: These transfer definitions are imported into the AVLC where they are subsequently used by the transfer protection function.

Table: JOURNEY CONNECTION (EINZELANSCHLUSS) (432)					
Key	Relation attributes	Data type	Value range	Required for	Description
P1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P2	CONNECTION_ID (EINAN_NR)	decimal 5)	1 - 32764	AVLC	Unique number for a transfer definition
	CONNECTION_NAME (ANSCHLUSS_NAME)	char(40)	ISO 8859-1	AVLC	Text to name the transfer
	PRIORITY (ANSCHLUSS_GRUPPE)	char(6)	ISO 8859-1		Free grouping of transfers to reflect priorities.
	CONTROL CENTRE CODE (LEITSTELLENKENNUNG)	decimal (3)	1 - 255 (0)	AVLC	Identification of the AVLC Control Centre with which information is exchanged (see VDV Recommendation 453). If the feeder vehicle belongs to the third-party control centre, the attribute has a value > 0. The value of this attribute defines which combination of attributes are read: If the Control Centre Code = 0 the following attributes are read: • FEEDER_LINE_NO (ZUB_LI_NR) • FEEDER_DIRECTION (ZUB_LI_RI_NR) • FEEDER_STOP_NO (ZUB_ORT_REF_ORT) If Control Centre Code (Leitstellenkennung) > 0 • LINE_ID (LinienID), • DIRECTION_ID (RichtungsID) • and ConnectionLinkRef (ASBID) are

					supplied. Non-supplied attributes are assumed to be 0 or empty.
	FEEDER_LINE_NO (ZUB_LI_NR)	decimal (6)	1 - 999		Route number of the feeder (VDV Data Model 4.1: LINE_NO (LI_NR) Identifier of the transport supply as a route or direction) 2)
	FEEDER_DIRECTION (ZUB_LI_RI_NR)	decimal (3)	1 - 2 (0)		Direction of the feeder route (VDV Data Model 4.1: DIRECTION (LI_RI_NR) identifies the route direction) 2)
	FEEDER_STOP_NO (ZUB_ORT_REF_ORT)	decimal (6)	>0 1)		Point where passengers alight from a feeder vehicle to change to the receiver vehicle.
	FEEDER_POINT_TYPE (ZUB_ONR_TYP_NR)	decimal (2)	>0, NULL	Optional	Location type of the feeder point.
	FEEDER_POINT_NO (ZUB_ORT_NR)	decimal (6)	>0, NULL	Optional	Point where the passengers leave the feeder vehicle to catch the connection (transfer).
	FROM_STOP_NO (VON_ORT_REF_ORT)	decimal (6)	>0, NULL	Optional	Stop, from where the feeder vehicle approaches the feeder stop
	LINE_ID (LinienID)	char(6)	ISO 8859-1		Identifier of the feeder route; must be supplied instead of FEEDER_LINE_NO if the feeder belongs to a 'third-party' AVLC system (control centre).
	DIRECTION_ID (RichtungsID)	char(6)	ISO 8859-1		Identifier of the feeder route; must be supplied instead of FEEDER_DIRECTION (ZUB_LI_RI_NR) if the feeder belongs to a 'third-party' AVLC system (control centre).
	ConnectionLinkRef (ASBID)	char(10)	ISO 8859-1		Connection protection area identification It must be agreed by the participating control centres and is only supplied if the feeder belongs to a ‘third-party’ AVLC system (control centre) Number of a systematic transfer
	FETCHER_LINE_NO (ABB_LI_NR)	decimal (6)	1 - 999	AVLC	Route number of the receiver (VDV Data Model 4.1: UMS_Z_NR

					identifies the route to which the passengers transfer.) 2)
	FETCHER _ DIRECTION (ABB_LI_RI_NR)	decimal (3)	1 - 2 (0)	AVLC	The direction determines the destination of the routes (VDV DM 4.1: UMS_Z_RI identifies the route direction of the destination route) 2)
	FETCHER_STOP_NO (ABB_ORT_REF_ORT)	decimal (6)	>0 1)	AVLC	Stop at which the passengers board the receiver vehicle of the transfer (VDV Data Model 4.1: UMS_Z_ORT is the location number of the point on the trip from which the journey continues after the transfer) 2)
	FETCHER_POINT_TYPE (ABB_ONR_TYP_NR)	decimal (2)	>0, NULL	Optional	Location type of the receiver point.
	FETCHER _POINT_NO (ABB_ORT_NR)	decimal (6)	>0, NULL	Optional	Location at which the passengers board the receiver vehicle of the transfer.
	TO_STOP_NO (NACH_ORT_REF_ORT)	decimal (6)	>0, NULL	Optional	Stop via which the receiver vehicle continues it's journey

1) IN AVLC operation, it is necessary to check whether the entire range of values can be used. Many transport authorities use equipment that only allows a range of 1 - 9999.

2) Thus the attribute names are different from what might have been expected from the first part of this recommendation or VDV Data Model 4.1.

Note: With the feeder or receiver stopping point, it is possible to include only those feeder and receiver vehicles that serve the exact stopping point. If this attribute is not specified, all feeders and receivers of the given route/direction are used.

With the from and to stops, it is possible to select a more accurate path for the feeder or receiver on the given route/direction. In this situation, the system only considers the patterns that travel via the 'from' stop to the feeder or travel away from the receiver via the 'to' stop. If these attributes are not specified, all feeder and receiver patterns for the given route/direction are taken into consideration.

10.2 INTERCHANGE (REC_UMS) (232)

Description: The protection of transfers may be restricted to certain day types and times. A transfer definition therefore, can have different validities. Depending on the time of day, the transfer monitoring can use different transfer and delay times.
(Description of transfer possibilities or systematic connections)

Table: INTERCHANGE (REC_UMS) (232)					
Key	Relation attributes	Data type	Value range	Required for	Description
P1	BASE_VERSION (BASIS_VERSION)	decimal (9)	>0	AVLC	Label of the general version
P2	CONNECTION_ID (EINAN_NR)	decimal (5)	1 - 32764	AVLC	Unique number for a transfer definition
P3	DAY_TYPE_NO (TAGESART_NR)	decimal (3)	1 - 999	AVLC	Day type label
P4	VALIDITY_START_TIME (UMS_BEGINN)	decimal (6)	0 - 129599	AVLC	Time in seconds after midnight for the validity start time within a day type.
P5	VALIDITY_END_TIME (UMS_ENDE)	decimal (6)	0 - 129599	AVLC	Time in seconds after midnight for the validity end time within a day type.
	INTERCHANGE_STANDARD_DURATION (UMS_MIN)	decimal (5)	0-65532	AVLC	Minimum transfer time for a transfer connection. Minimum changeover time in seconds for a passenger to get from the feeder stopping point to the receiver stopping point.

	INTERCHANGE_MAXIMUM_DURATION (UMS_MAX)	decimal (5)	0-65532	AVLC	Maximum transfer time for a transfer connection. Maximum time in seconds, in which the passenger can reasonably be assumed to make the transfer (inclusive of waiting time), so that the transfer can still be regarded as a transfer. This attribute is used to form transfer pairs.
	MAXIMUM_WAIT_TIME (MAX_VERZ_MAN)	decimal (5)	0-65532	AVLC	Maximum timetable deviation in seconds that is allowed for the receiver as a consequence of a transfer protection decision taken manually by a supervisor.
	MAXIMUM_WAIT_TIME_AUTO (MAX_VERZ_AUTO)	decimal (5)	0-65532	AVLC	Maximum timetable deviation in seconds that is allowed for the receiver vehicle as a consequence of an automated transfer protection decision by the AVLC system. If this value is exceeded, it must be acknowledged by the dispatcher so that the transfer continues to be monitored.

Use of font types for attribute names:
Normal = the name is the same as in VDV Data Model 4.1

Italics = not in VDV Data Model 4.1

Underlined = VDV453

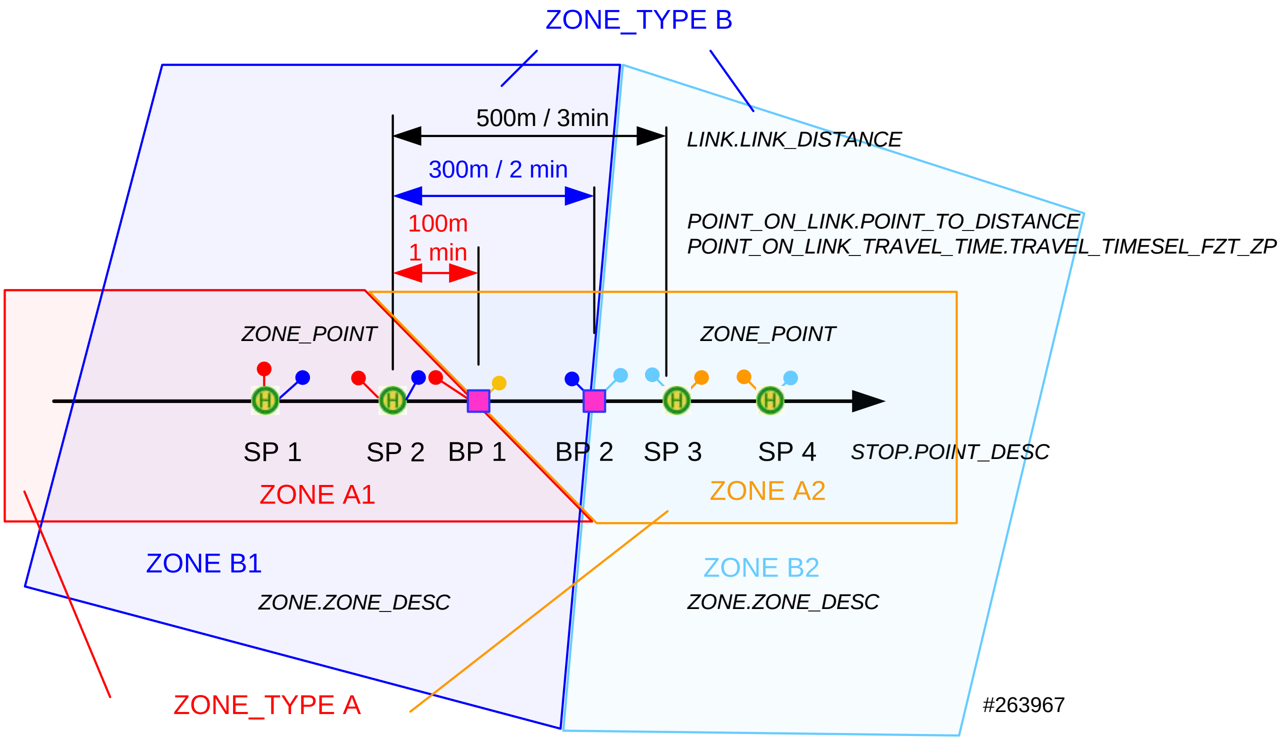
11 Extension of the zones

The aim of the extension to the Standard VDV 452 Route Network / Timetable Interface described in this chapter is to model the zones or administrative units.
Zones are spatially connected zones that are used in the planning programs to calculate the services provided by the various administrative units. The VDV 452 interface must be extended before this planning data can also be used for AVLC operation. This modelling is not intended for the depiction of tariff models.

The interface is extended with the ZONE (FLAECHEN_ZONE) table. This table lists all zone types and the various operational areas associated with each type. The operational areas of an zone type are connected and must not overlap.
The stops, depot points and border points are assigned to the zones in which they lie. The ZONE_POINT (FL_ZONE_ORT) table is used for this allocation.

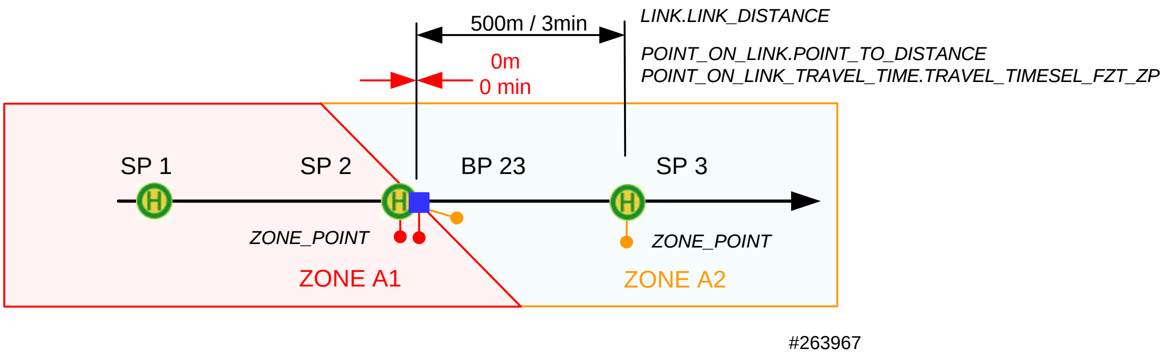
If 2 consecutive stopping points and/or depot points of a route are located in different zones, a border point must always be placed between them. This border point is saved in the STOP (REC_ORT) table with POINT_TYPE (ONR_TYP_NR) = 7 (border point). This border point is then assigned to the area in ZONE (FLAECHEN_ZONE) via the ZONE_POINT (FL_ZONE_ORT) table. If the border point forms the intersection between 2 ZONEs (FLAECHEN_ZONE), it is always assigned to both ZONEs (FLAECHEN_ZONE).
The distance between the stopping point and the border point is stored in the POINT_ON_LINK (REC_SEL_ZP) table and POINT_TO_DISTANCE (SEL_ZP_LAENGE) attribute. The distance is always specified with reference to the start stopping point of the pattern. The travel time from the stopping point to the border point is stored in the POINT_ON_LINK_TRAVEL_TIME (SEL_FZT_FELD_ZP) table and TRAVEL_TIME (SEL_FZT_ZP) attribute.

The picture below shows 4 stopping points (SP 1 to SP 4) which lie within 2 zone types and a total of 4 zones. Stopping points SP1 and SP2 are assigned to zones A1 and B1. Stopping points SP3 and SP4 are assigned to zones A2 and B2. The path runs from SP1 in the direction of SP4. BP1 describes the boundary between zones A1 and A2 and is therefore assigned to both zones. Border point BP1 lies 100m after stopping point SP2 on the path to SP3. The vehicle requires 1 minute to travel to the border point. BP2 describes the boundary between zones B1 and B2 and lies 300m after stopping point SP2 on the path to SP3. The vehicle requires 2 minutes to travel to this border point.



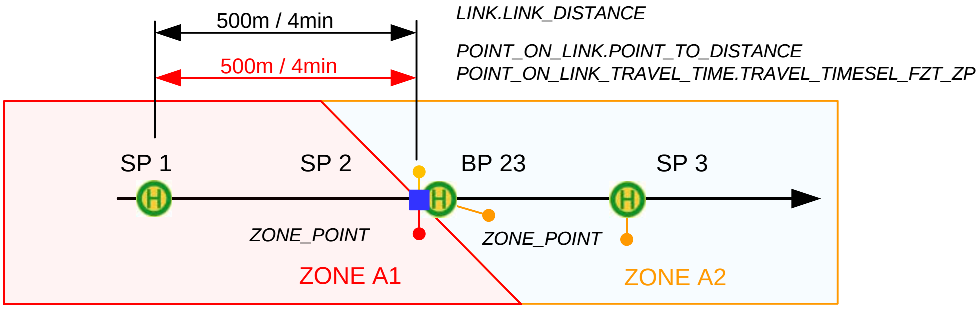
If the border point coincides exactly with a stopping point, there are a possible 4 scenarios to consider. In all the scenarios depicted below, the border point sits directly at the stopping point and border point BP23 is assigned to zones A1 and A2.

The route path cuts the boundary and the stopping point belongs to zone A1:



In this situation, border point BP23 is located on the path between SP2 and SP3 with POINT_TO_DISTANCE (SEL_ZP_LAENGE) = 0.

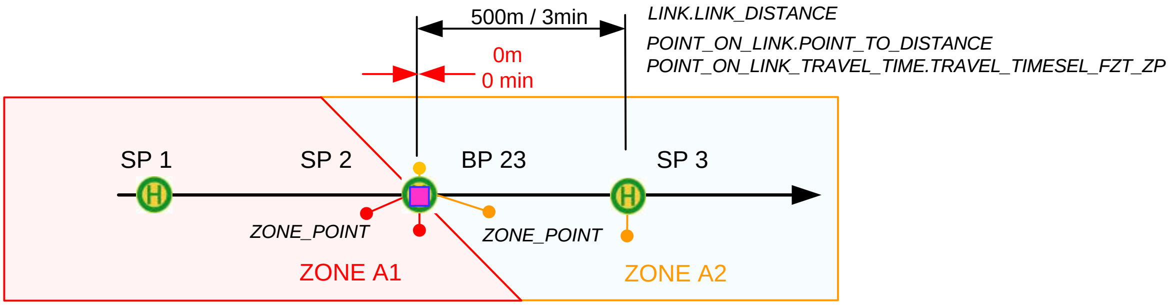
The route path cuts the boundary and the stopping point belongs to zone B1:



#263967

In this situation, border point BP23 is located on the path between SP1 and SP2 with POINT_TO_DISTANCE (SEL_ZP_LAENGE) = distance between SP1 and SP2.

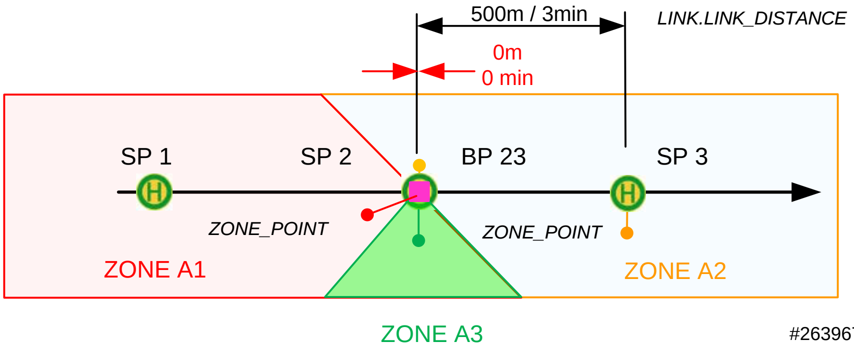
The route path cuts the boundary and the stopping point belongs to zones A1 and A2:



#263967

In this situation, border point BP23 is located on the path between SP2 and SP3 with POINT_TO_DISTANCE (SEL_ZP_LAENGE) = 0.

The route path skirts the boundary but doesn't cut across it. The stopping point belongs to A3 but not A1 or A2:



#263967

In this situation, border point BP23 is located on the path between SP2 and SP3 with POINT_TO_DISTANCE (SEL_ZP_LAENGE) = 0.

12 European Standards

With regard to the transmission of information relating to the route network and timetable, the 'NeTEx' interface has been developed on the basis of the European 'Transmodel' data model.

12.1 NeTEx

VDV Recommendation 452 was tabled by the CEN task force when creating NeTEx. This means that all data elements of VDV Recommendation 452 have a corresponding element in NeTEx.

The **objectives** of NeTEx are as follows:

NeTEx is dedicated to the exchange of scheduled data (network, timetable and fare information). It is based on Transmodel V5.1 (EN12986), IFOPT (CEN/TS 00278207) and SIRI (CEN/TS 1553-1 to 3 and TS 00278181-4 and 5) and supports the exchange of information of relevance for passenger information about public transport services and also for running Automated Vehicle Monitoring Systems (AVMS, also known as AVLC (Automatic vehicle location & control system)).
NOTE
Many NeTEx concepts are taken directly from Transmodel and IFOPT; the definitions and explanation of these concepts are extracted directly from the respective standard and reused in NeTEx, sometimes with adaptations in order to fit the NeTEx context.

Although the data exchanges targeted by NeTEx are predominantly oriented towards provisioning passenger information systems and AVMS with data from transit scheduling systems, it is not restricted to this purpose and NeTEx can also provide an effective solution to many other use cases for transport data exchange.

There is a **comparison of VDV452 and NeTEx** and an example of usage on the VDV website at WWW.vdv.de under Technology/Projects or via http://www.vdv.de/netex.aspx.

12.2 Comparison of VDV452 German - English - Transmodel

Content of TRANSMODEL

Contrary to the German VDV Data Model, TRANSMODEL is not a *logical data model*, which can be used directly as the starting point for the definition of a database model of a specific application, but a *conceptual data model* with a focus on the semantic description of a realistic excerpt according to methodical rules and pre-specified descriptive elements.

The data dictionary underlying the TRANSMODEL is restricted to a definition of the conceptual entities and their most important attributes (mainly in connection with identification) as well as the representation of the logical network of relationships between the individual entities. There is no definition of the data types or lengths, units or value ranges of the attributes, nor does it take into account any application-specific optimisation of the data structures in the context of performance or memory requirements. The challenge was to offer transport authorities and developers a uniform benchmark for semantic data modelling.

There is a downloadable table on the VDV website at WWW.VDV.DE, which compares the German VDV Recommendation 452, it's English translation and the data objects of TRANSMODEL.

The corresponding tables of the VDV Data Model can therefore be construed as realistic implementations of the conceptual TRANSMODEL entities

The European standard EN12896, known as “Transmodel” (from EN12896, “Foreword”)

Transmodel 5.1 is a reference standard which provides a conceptual data model for use by organisations with an interest in information systems for the public transport industry.

As a reference standard, it is not necessary for individual systems or specifications to implement Transmodel. However, it must be possible to describe (for those elements of systems, interfaces and specifications which fall within the scope of Transmodel):

The aspects of Transmodel that they have adopted and the aspects of Transmodel that they have chosen not to adopt.

For an organisation wishing to specify, acquire and operate information systems, Transmodel may be distilled, refined, or adapted to form a comprehensive data model for the organisation, or specific data models for database design or interface specification.

For an organisation wishing to design, develop and supply information systems, Transmodel may be distilled, refined, or adapted to form a comprehensive data model for the product suite.

13 Possible future developments and options

This appendix is provided to help avoid contradictions between developments for specific projects. The additional data elements listed here must not under any circumstances be used in other projects with different meanings.
If a data element is required in another project and there is already an element with the same meaning listed here, it is essential that the label, format and length stated in this appendix is used.

All users of this VDV Recommendation are requested to report any customised extensions immediately to VDV, including relations, value range changes or additional data elements.
These are published at WWW.VDV.DE under VDV Projects – VDV Data Model and will be included in a later version of this recommendation.
This includes, for example, two alternative data storage concepts for the calendar ('differential validity model' and 'calendar extension').