# DINO Exchange Format

# Version 2.3

## DINO Exchange Format Documentation

## Version history

| Document version | Date | Name | Reason for change |
|---|---|---|---|
| 1.0 | May 2001 | | Viking Study MIP01 - Interim Status |
| 1.1 | Oct 2001 | | Viking Study MIP01 - Final Standings |
| 1.2 | March 2002 | | MIP01 - Level S |
| 1.3 | Jan 2003 | | MIP02 - Level V |
| 1.4 | Nov 2005 | | Extension to include VDV 454, operating prohibitions and use indicators for the crates |
| 1.5 | 18.6.2007 | M. Steel | Document structure recreated. No changes to the specification itself. |
| 1.5.1 | 22.7.2009 | M. Steel | Extension with stop-related hinw_str.din, see chapter 5.6.5<br>Please note that Translation errors in older versions with decisive effects in the lid_course.din table, see chapter 5.5.11<br>Inclusion of the optional table branch.din, see chapter 5.5.9 |
| 1.6 | 12.01.2010 | M. Steel | Extension of the tables for vehicle destination texts vehicle_destination_text.din and trip_vdt.din.<br>IFOPT attribute extension for the tables rec_stop, rec_stop_area and rec_stopping_points |
| 1.6.1 | 20.02.2012 | M. Steel | Graphical explanation of service_restriction **Fehler! Verweisquelle konnte nicht gefunden werden.** |
| 1.7 | 22.04.2013 | E. Panholzer | Extension with means of transport and transfer matrix<br>means_of_transport, transfer_matrix |
| All enhancements from 1.7 affect the export from DIVA4 | | | |
| 1.7 | 14.05.2013 | E. Panholzer | Additions |
| 1.7 | 02.08.2013 | F. Twaroch | Additions |

| Document version | Date | Name | Reason for change |
|---|---|---|---|
| 2.0 | 20.01.2014 | F. Twaroch | Correction: TIMETABLE_PERIOD has 4 digits in all DINO tables<br><br>Extension of restriction in service_restriction.din from char(4) to char(5).<br><br>Coordinate format for stops and climbs extended from 7 to 12 digits, thus also supporting WGS 84 coordinates with sufficient accuracy<br><br>Table means_of_transport is renamed means_of_transport_desc<br><br>Rec_trip: Ext_Trip_Key column added, not used in DIVA<br><br>Added tables for transferring partial links (rec_links.din) constraint points (rec_link_force_point.din) and partial link polygons (rec_link_geometry.din).<br><br>Table for tariff zone neighbours was introduced (rec_neighbour_fare_zone.din)<br><br>Tables for the import of connection definitions (identical to VDV 452) were introduced, (Einzelanschluss.din, rec_ums.din)<br><br>Stops (rec_stop.din) - Extension by the field "STOP_NAME_WITHOUT_LOCALITY".<br><br>Areas have been extended by coordinates, level and type, short name and long name, rec_stop_area.din - extension by the optional fields STOP_AREA_POS_X, STOP_AREA_POS_Y, STOP_AREA_SHORT_NAME, STOP_AREA_LONG_NAME, LEVEL, STOP_AREA_TYPE. |
| 2.0 | 20.01.2014 | F. Twaroch | stop_footpath.din - Extension by the "PROPERTY" column<br>rec_neighbour_fare_zone.din - Introduction of a new table for the transfer of fare zone neighbours<br>rec_lin_ber.din - Extension by the fields "VALID_FROM", "VALID_TO" for modelling the validity of a line<br>trip_vdt.din - Adaptation of the import behaviour, and the table, only destination text changes are saved and no longer the destination text of each stop in the route<br>rec_trip.din - Extension to include train types, introduction of "TRAIN_CATEGORY_SHORT_NAME".<br>notice.din - Introduction of the fields "CONTENT_TYPE" and "DISPLAY_TYPE" - Used to distinguish between different types of notices and to control when notices should be issued by the EFA.<br>rec_connection.din - change of meaning, this table is only used for the transmission of connection links anymore<br>Individual connection.din - Introduction of a table for the transmission of connection definitions<br>rec_ums.din - Introduction of a table for the transmission of connection monitoring<br>rec_links.din - Introduction of a table for the transmission of partial routes<br><br>rec_link_force_point.din - Introduction of a table for the transmission of constraint points<br>rec_link_geometry.din - Introduction of a table for the transmission of georeferenced partial routes |
| 2.0 | 25.04.2014 | M. Steel | Chapter 3 introduced<br>In table **Fehler! Verweisquelle konnte nicht gefunden werden.** the attribute LINE_NR is added to reflect the distinction between global and line-related traffic restrictions. |
| 2.0 | 23.05.2014 | F. Twaroch | The occasionally used abbreviation NO for number was abolished and replaced by the VDV 452 compliant abbreviation NR.<br><br>Section Annex defined and extended by list of permissible train types. |
| 2.0 | 08.07.2014 | F. Twaroch | Inclusion of relations for the support of entrepreneurs<br>operator.din<br>operator_branch_office.din<br><br>The tables rec_lin_ber.din and rec_trip.din have been extended by a foreign key OBO_NR. |
| 2.0 | 25.08.2014 | F. Twaroch | Inclusion of relations to support journey-related attributes:<br>set_trip_attribute.din |

| Document version | Date | Name | Reason for change |
|---|---|---|---|
| | | | The tables rec_lin_ber.din and rec_trip.din have been extended by a foreign key OBO_NR. |
| 2.0 | 12.03.2015 | F. Twaroch | The PLACE maintained in Stop_point will become obsolete in the future, instead a global location ID will be introduced. Until then, the newly introduced PLACE_ID field is to be filled by the sub-location number.<br><br>Stop points stop.din, stop_area.din, stop_point.din are extended by GIS traffic flags. |
| 2.0 | 12.5.2015 | F. Twaroch | Extension of DAY_TYPE_NR decimal (9) and DAY_ATTRIBUTE_NR decimal (5). |
| 2.0 | 28.5.15 | F. Twaroch | Extension of VEH_TYPE_NR to decimal (3) extended , number range VDV compliant to 1... 252 expanded. |
| 2.0 | 29.05.2015 | F. Twaroch | Supplement - Explanation: The TRIP_EXT_KEY parameter was included in the trip.din table for a third party provider, but this is not supported by DIVA. |
| 2.0 | 01.06.2015 | F. Twaroch | Extension with relations to support the transmission of fare information (fare_zone_transition.din, fare_zone_transition_point.din) and coordinate systems (coordys).<br><br>In the table notice.din there is only one text field for notice texts. This has a length of 1000 characters.<br><br>Trip_stop_time.din: Redundant attributes removed<br><br>The IFOPT attribute has been renamed GLOBAL_ID.<br><br>REF_STOP_NR and REF_STOP Name are no longer supported in the stop.din and stop_point.din tables. |
| 2.0 | 18.06.2015 | F. Twaroch | Introduction of the table train_category.din to manage train categories.<br><br>The table round_trip.din (vehicle rounds) was renamed to vehicle_block.din. Operational data is not yet supported by the DIVA interface DINO 2.0. |
| 2.0 | 15.07.2015 | F. Twaroch | The tables for the connection definitions have been given the more meaningful DIVA names: interchange_definition.din and interchange_validity.din<br><br>Several new DINO relations attribute.din, stop_attribute.din, stop_area_attribute.din, stop_point_attribute.din, and line_attribute.din to support custom attributes have been introduced.<br><br>The relations stop_area.din and stop_point.din have been extended by validity attributes.<br><br>Hyperlinks corrected |
| 2.0 | 17.11.2015 | F. Twaroch | Extension of the interchange_definition.din to include areas for feeders and drop-offs. |
| 2.0 | 23.11.2015 | F. Twaroch | Extension of connection.din by type specification.<br><br>The column for vehicle_change in connction.din is obsolete and has been removed. |
| 2.1 | 10.12.2015 | S. Engelhardt | Extensions for the exchange within the framework of DELFIplus |
| 2.1 | 09.02.2016 | F. Twaroch | Extension trip_purpose.din, the relation trip.din gets a new column PURPOSE_NR |

| Document version | Date | Name | Reason for change |
|---|---|---|---|
| 2.1 | 11.5.2016 | F. Twaroch | Extensions Connection definition:<br><br>1. CONNECTION_NR is 10-digit<br><br>2. PRIORITY does not exist in interchange_definition.din but in interchange_validity.din.<br><br>3. CONNECTION_NAME is 100 characters<br><br>4. feeder line, drop-off line and direction must be included in the key, as there can be several lines per connection definition number.<br><br>5 The texts for the foreign line identifier are 20 characters each<br><br>6. PROTECTION_TYPE also needs the value <empty> if SECURED_INTERCHANGE = 0 not secured.<br><br>7. DAY_TYPE_NR must be DAY_ATTRIBUTE_NR, as no DAY_TYPE_NR is specified for the affected journeys.<br><br>8. for interchange_validity the key must be CONNECTION_NR + VALIDITY_START_TIME<br><br>9. the times in interchange_validity are all 6 digits.<br><br><br>Value range extensions:<br><br>1. GLOBAL_ID: Value range increased to 100 characters as two keys with 50 characters each are combined, remark: GLOBAL_ID of the line + REAL_TIME_ID of the trip (real time ID)<br><br>2. VEH_TYPE_NR: Value range extended to 8 characters because RBL number in DIVA is 8 characters long. This affects all tables that contain this attribute. |
| 2.1 | 25.05.2016 | F. Twaroch | The realtion stop.din was extended by a transfer quality (INTERCHANGE_QUALITY) |
| 2.1 | 1.6.2016 | F. Twaroch | Train number extended from 5 to 8 digits |
| 2.1 | 13.06.2016 | F. Twaroch | Correction STOP_AREA_NR can be 0 if it represents a zero range. The zero area is a pseudo area to define stops with climbs but without areas. However, the zero area cannot carry any properties. |
| 2.1 | 29.08.2016 | F. Twaroch | Unproductive journeys in route.din<br><br>Stops can be marked as unproductive in routes, which allows them to be "filtered out" in DINO exports.<br><br>The attribute value "5 = no passenger transport" for the attribute STOPPING_POINT_TYPE in the relation route.din was newly introduced. |
| 2.1 | 23.11.2016 | F. Twaroch | Correction GLOBAL_ID was erroneously generally specified as 100 characters long, but is only 50 characters long. Only the GLOBL_ID of the trip (trip.din) is specified with a length of 100 characters. |
| 2.1 | 28.12.2016 | F. Twaroch | The INTERCHANGE_TYPE column was added to the stop.din relation. |
| 2.1 | 24.02.2017 | F. Twaroch | Addition: The attributes BIKE_RULE (line.din) and BIKE_ALLOWED (trip.din) were introduced for the DELFI Plus project to account for bike sharing and bike routing, but are not supported by the current DIVA 4 DINO import. The use of the parameters must be clarified with a MENTZ project manager. |
| 2.1 | 03.04.2017 | F. Twaroch | Supplementary explanations for connection definitions in DINO, concerns the relations interchange_definition.din and interchange_validity.din. |
| 2.1 | 23.08.2017 | T.Jakubicka | TYPE Description added to attribute.din |
| 2.1 | 08/09/2017 | F. Twaroch | Tariff zones have been extended from 4 to 5 digits and can now also carry a validity. Concerns the relations fare_zone.din, neighbour_fare_zone.din, fare_zone_transition .din, fare_zone_transition_point.din |

| Document version | Date | Name | Reason for change |
|---|---|---|---|
| | | | Corrections and further explanations: |
| | | | ATT_SHORT_NAME extended from 3 to 12 digits |
| | | | Means_of_transport_desc.din extended by a column for the textual description of the transfer means of transport |
| | | | Extension of the documentation on service prohibitions, service_constraint.din. |
| 2.1 | 6.8.2018 | S. Engelhardt | Correction ATT_TPYE in attribute.din<br>Clarification Uniqueness in trip.din |
| 2.1 | 25.09.2018 | S. Engelhardt | Corrector minimum requirements (not trip_stop_time.din) |
| 2.1 | 27.12.2018 | M. Steel | New notice.CONTENT_TYPE defined, see notice.din chapter 5.6.4.<br><br>5 .. R-Bahn (R-train)<br><br>6 .. driver message (driver text)<br><br>7 .. facillity (offer)<br>8 .. fare code (tariff code) |
| 2.1 | 17.04.2020 | F. Twaroch | Additions - description minimum scope, operator.din and operator_branch_office.din have been included in the overview and are optional tables |
| 2.1 | 29.09.2020 | F. Twaroch | Correction to link.din: STOPPING_POINT_NR is also part of the key<br><br>Clarification of the mandatory fields in stop_attribute.din and stop_point_attribute.din. The stop number or platform number must be provided. |
| 2.1 | 22.01.2021 | F. Twaroch | Extension of vehicle_destination_text.din by columns for the interior display (VDV_TEXT_INNEN 1 to 4). |
| 2.1 | 08.03.2021 | F. Twaroch | Extension of the line.din to include special tariff features: LINE_SPECIAL_FARE. |
| 2.2 | 08.11.2021 | P. Arnsberger | Table version.din gets new column DINO_FORMAT |
| 2.2 | 08.11.2021 | P. Arnsberger | Table line.din receives new columns LINE_SHORT_NAME, LINE_SUFFIX, LV_VERSION |
| 2.2 | 08.11.2021 | P. Arnsberger | Table stop_additional_name.din gets new key columns ;BRANCH_NR;TMOT_NR;NAME_TYPE;SEQUENCE_NO |
| 2.2 | 08.11.2021 | P. Arnsberger | Table route.din stop types and description of the field STOPPING_POINT_TYPE extended.<br>Similarly, the SERVICE_INTERDICTION_CODE field in table service_constraint.din has been extended to include demand stops and operational stops. |
| 2.2 | 08.11.2021 | P. Arnsberger | Error in table trip.din fixed: Field ROUND_TRIP_NR is not part of the key, as rotations are currently not part of the DINO export until version 2.3. |
| 2.2 | 15.11.2022 | P. Arnsberger | Table trip.din: GLOBAL_ID - Field length extended from Char(100) to Char(128). The background is the expected standardisation of the Global ID at EU level. |
| 2.2 | 15.11.2022 | P. Arnsberger | Table stop.din, field GLOBAL_ID: Field length increased from Char(50) to Char(128) from version 2.2. |
| 2.2 | 15.11.2022 | P. Arnsberger | More than two directions / direction numbers are now possible. When importing into DIVA, these must be assigned to outward/backward. |
| 2.2 | 23.11.2022 | L. Schwartz | Chapter 6.1 Annexes > List of permitted train types removed. This was a relic from DIVA 3. The train types can be stored flexibly in a DB table in DIVA 4. |
| 2.2 | 23.11.2022 | L. Schwartz | Formatting adjusted, link to jump to "List of all relations" in each subchapter removed. Table 3.1 Minimum scope and capability of various products removed. |

# DINO 2.3

| Document version | Date | Name | Reason for change |
|---|---|---|---|
| 2.2 | 25.11.2022 | L. Schwartz | Note on the meaning stop. IS_RESPONSIBLE_STOP added: Indicates that the exporting subnetwork is responsible for this stop. |
| 2.2 | 25.11.2022 | L. Schwartz | Notes on encoding / character set added to chapter **Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.**. Correction: ANSI (instead of ASCII) has been the standard for a long time. |
| 2.2 | 08.12.2022 | L. Schwartz | Notes on the handling of permitted line breaks added to chapter 4 and to table notice.din > NOTICE_TEXT. |
| 2.3 | 08.11.2021 | P. Arnsberger | Table route gets new columns PRINT_FLAG; PRINT_FLAG_SSTT and OPTION_FLAG |
| 2.3 | 18.11.2021 | P. Arnsberger | Write error in table train_category.din BASIS_VERSION -> VERSION |
| 2.3 | 26.11.2021 | P. Arnsberger | To identify a trip, LINE_NR and TRIP_ID are sufficient, therefore the key information in trip_vdt and notice_str are incorrect, cf. also trip_stop_time, service_constraint and trip_part |
| 2.3 | 13.12.2021 | P. Arnsberger | New column STREET_ACCESS_HEIGHT in stop_point |
| 2.3 | 21.04.2022 | W. Düx | Draft DELFI attributes of the vehicle types |
| 2.3 | 04.05.2022 | S. Engelhardt | VEH_DELFI_TYPE added |
| 2.3 | 13.06.2022 | W. Düx | Assignment of connecting objects such as lifts, escalators, stairs, ramps to footpaths (new table "stop_footpath_asset") |
| 2.3 | 15.11.2022 | L. Schwartz | Tables line.din and notice.din: Values of the DISPLAY_TYPE field adapted to DIVA4 and extended. |
| 2.3 | 15.11.2022 | L. Schwartz | Table stop.din: Length of the STOP_NAME and STOP_NAME_WITHOUT_LOCALITY fields increased from 50 to 255 characters. |
| 2.3 | 15.11.2022 | L. Schwartz | ER diagrams and tab list added |
| 2.3 | 15.11.2022 | L. Schwartz | Columns FK for foreign key relations / references in other DINO tables added. |
| 2.3 | 13.03.2023 | L. Schwartz | Clarification of the handling of validity of the timetable version (chapter version.din) |
| 2.3 | 13.03.2023 | S. Engelhardt | Optional column stop.FARE_PROVIDER_CODES inserted. Background: Within the DELTa project (BMVI & DELFI) (and presumably later in the regular DELFI data exchange), it must be possible to transport a so-called tariff knowledge for each stop. These stop markings make it easier to break connections by tariff in order to determine fares. |
| 2.3 | 24.03.2023 | L.Schwartz | • Name correction in table Vehicle_Destination_Text: VDT_TEXT_INNEN[1-4] to VDT_TEXT_INTERIOR[1-4]<br>• Name correction in table Trip: ROUND_TRIP_NR to ROUND_TRIP_ID<br>• Correction in table Trip_vdt: Addition of STOP_NR and STOPPING_POINT_NR columns - these were previously missing from the specification.<br>• Correction in table Vehicle_Block: Addition to the DAY_ATTRIBUTE_NR column. → this replaces DAY_TYPE_NR since DINO 2.2<br>• Correction in table Line_Suppression: VERSION column added |
| 2.3 | 03.04.2023 | M. Steel | Approved by |

## Table of Contents

# 1 Introduction

The DINO exchange format is based on the VDV-DIVA exchange format which, in turn, is based on the German ÖPNV VDV standard, version 5.0 or 5.1.

At the request of many customers, the DINO data documentation has been revised considerably. In recent years, some extensions to the interface were also implemented and introduced, some of which deviate from the VDV standard. One main difference from the VDV standard is the way trip times and day types are modelled. A test data set to illustrate the changes in the definitions is provided together with this documentation.

Some inconsistencies in the interface were also cleaned up. The short designation NO which was occasionally used for "number" was therefore consistently replaced with the VDV 452-compliant abbreviation NR. Field lengths were extended so that they apply uniformly to all tables in the specification (e.g. *TIMETABLE_PERIOD*).

Due to the many changes compared to previous versions, a new version number was assigned to the interface. Future DINO exports will be made available with version number 2.x. DINO 2.x data is not backward compatible with DINO 1.x.

## 2   Overview of all relations

The following table shows all the tables supported by the DINO format. From version DINO 2.0 onwards, attention was also paid to naming the tables more uniformly. Many tables were therefore renamed compared to the previous version. The new table names are listed here:

| Category | from DINO 2.0 | up to DINO 1.7 |
|---|---|---|
| **General specification** | character.set.din | - |
| **Calendar dates** | version.din | set_version.din |
| | day_type.din | set_day_type.din |
| | day_attribute.din | set_day_attribute.din |
| | day_type_2_day_attribute.din | day_type_2_day_attribute.din |
| | day_type_calendar.din | calendar_of_the_company.din |
| | service_restriction.din | service_restriction.din |
| **Location data** | stop.din | rec_stop.din |
| | stop_area.din | rec_stop_area.din |
| | stop_point.din | rec_stopping_points.din |
| | stop_footpath_din | rec_footpath.din |
| | stop_additional_name.din | rec_additional_stopname.din |
| | stop_alias_placename.din | rec_alias_placename.din |
| | coordsys.din | - |
| **Fare data** | fare_zone.din | - |
| | neighbour_fare_zone.din | rec_neighbour_fare_zone.din |
| | fare_zone_transition.din | - |
| | fare_zone_transition_point.din | - |
| **Mode of transport** | means_of_transport_desc.din | means_of_transport_desc.din |
| **Interchange times** | transfer_matrix.din | transfer_matrix.din |
| **Vehicle types** | vehicle_type.din | set_vehicle_type.din |
| | vehicle_type_delfi_attr.din | - |
| | vehicle_door_delfi_attr.din | - |
| **Operator** | operator.din | - |
| | operator_branch_office.din | - |
| | depot.din | set_depot.din |
| **Operating branches** | branch.din | branch.din |
| **Line, network, operational data** | timing_pattern.din | lid_travel_time_type.din |
| | route.din | lid_course.din |
| | trip_purpose.din | set_trip_purpose.din |
| | line.din | rec_lin_ber.din |

| | | |
|---|---|---|
| | vehicle_destination_text.din | vehicle_destination_text.din |
| | trip_vdt.din | trip_vdt.din |
| | train_category.din | - |
| | trip.din | rec_trip.din |
| | trip_stop_time.din | trip_stop_time.din |
| | vehicle_block.din | rec_round_trip.din |
| | line_suppression.din (from 2.1) | - |
| **Notes** | notice.din | notice.din |
| | notice_str,din | hinw_str.din |
| | service_constraint.din | service_interdiction.din |
| **Interchange definitions** | connection.din | rec_connection.din |
| | interchange_definition.din | - |
| | interchange_validity.din | - |
| **Sections and georeferenced data** | link.din | - |
| | link_geometry.din | - |
| | link_force_point.din | - |
| **User-defined attributes** | attribute.din | - |
| | stop_attribute.din | - |
| | Stop_area_attribute.din | - |
| | Stop_point_attribute.din | - |
| | Line_attribute.din | - |

## 3   Incremental data exchange

Some projects proposed an incremental data exchange. If this option is taken, the data elements that are to be exchanged must be agreed in advance. With DIVA 4 Release R15, an incremental DINO data delivery means a partial delivery of lines.

Here is an example to explain this. Take a timetable database consisting of a total of 10 lines within the timetable period. Now, for example, lines 1 and 3 change. It is now possible to transmit these two lines (1 and 3) in a separate incremental DINO data delivery. The created DINO export must contain all files that also contain the total delivery but only related to lines 1 and 3. Special consideration must be given if there are interchange definitions. If, in the same example, other lines (e.g. lines 5 and 7) have an interchange relationship with lines 1 and 3, these lines must also be included in the delivery. The incremental delivery then consists of lines 1,3,5,7 instead of the 10 lines of the total dataset.

The DIVA 4 DINO import can be configured to import DINO lines to an existing dataset as DIVA line versions or to overwrite existing line versions (see also notes on the validity of the DIVA line version for the relation version.din).

A DINO delivery dataset of individual lines must always contain all the relevant data elements that are dependent on these lines (stops, areas, platforms, service restrictions, notes, destination texts, etc.).

For incremental deliveries this means that the exporting system must transfer the minimum scope (see table) or the number of tables agreed for the project. If connection linkages are delivered, an incremental part delivery must include all the lines involved in the connection linkage.

Here, once again, is a list of the minimum DINO relations that are required:

- version.din
- day_type.din
- day_attribute.din
- day_type_2_day_attribute.din
- day_type_calendar.din
- service_restriction.din
- stop.din
- stop_area.din
- stop_point.din
- stop_footpath.din
- timing_pattern.din
- route.din
- line.din
- trip.din
- trip_stop_time.din
- notice.din
- service_constraint.din
- notice_str.din

If further optional relations are to be transferred in the incremental data delivery, these must also be exported / imported.... This must be decided before the data exchange takes place. If, for example, vehicle destination texts are to be exchanged, the vehicle_destination_text table must be supplied with the lines in the incremental data export. In the example above, only the destination texts of lines 1,3,5,7 need to be supplied, and not all 10 lines.

The line number must be kept persistent so that it can be assigned again during the next import. The DIVA 4 import uses an assignment table to assign the supplied DINO lines to the DIVA lines.

## 4   Notations

By default, the notation is based on the CSV specification[1].

The database tables are both imported and exported from DIVA in ANSI format (Windows-1252) by default. Also possible are, for example, ASCII or UTF8. The encoding is recorded in the table character_set.din. If this does not exist during import, the data set is interpreted in the Windows ANSI standard.

Each data table is created with the filename <filetablename>.din. The first line contains the table header in accordance with the VDV standard.

Separation of data fields should be done with a semicolon ";". other separators are possible. Data fields that are not served can be omitted by the delivery or import system (instead of default values or blanks).

Key:  Y = Yes, N = No, Opt = Optional

The selected separator must not be used in data fields unless it is enclosed by double apostrophes. Line breaks in text fields such as notice.din are allowed, but must also be enclosed in double apostrophes. The default separator of the DINO format is a semicolon. As in the CSV standard, there should be no separator at the end of the line.[2]

### 4.1   Data types

The following data types are used for DIVA and DINO in the documentation:

| Data type | Description |
|---|---|
| Decimal (x) | Decimal value, where x is the maximum number of digits |
| Char (x) | String, where x is the maximum number of net characters |
| Boolean | Logical type: 0 = FALSE / 1 = TRUE |
| ISO 8859-1 | |

*Table 1 Permitted data types*

---

[1] See https://csv-spec.org/

[2] "The last field in a record MUST NOT be followed by a comma. This results in an additional field with nothing in it." (https://csv-spec.org/)

# 5 Interface description

The DINO relations are described in the following chapters.

## 5.1 Code page
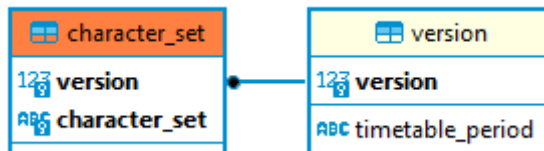


*Figure 1 ER diagram character encoding*

### 5.1.1 Character_set.din

ORACLE-compliant code page

Optional table.

This table defines the encoding used. S. ch. 4 Notations.

| Table: character_set | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Description | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Version | Basic version | Version.version |
| | Y | Obligation (mandatory) | CHARACTER_SET | char (20) | WE8ISO8859P1, EE8MSWIN1250, UTF8, ... | description of character set used | Character set used | - |

*Table 2 Character_Set*

A list of character sets that are supported by Oracle can be found at e.g.:
http://docs.oracle.com/cd/B28359_01/server.111/b28298/applocaledata.htm#i635016.

## 5.2 Calendar dates
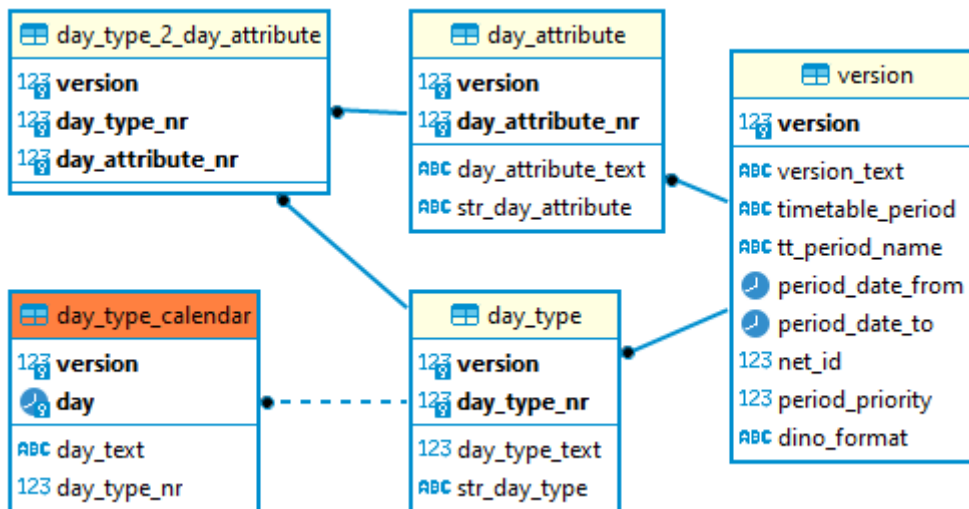


*Figure 2 ER diagram calendar data*

Trips are mapped to a calendar that indicates for each day whether or not the trip will take place. The necessary elements of the DINO data model are described in this section.

The relation version.din describes timetable periods as they exist in the DIVA data model. Timetable periods specify the validity range of a timetable. They are named with a freely definable key (e.g. s17 for summer timetable 2017 from 1.6.2017 to 26.9.2017). The key should provide an indication of the meaning. In principle, it is possible to work with any number of timetable periods at the same time. The lines do not have to have the same periodicity. There is only an Oktoberfest timetable for the lines that travel to the Wies'n. There may be overlaps between the timetable periods, in which case priority rules are defined. For example, the Christmas timetable overlaps with the winter timetable and yet there is no need to create a gap in the validity period of the winter timetable.

Other relations are used for defining service days. Service days are those days on which a trip takes place. In most timetable books, the timetables are presented separately according to the traffic days "Mondays to Fridays", Saturdays", Sundays and public holidays". For the railways, a "daily" representation in the traffic day is common.

The essential elements are day types or day type groups (in the DIVA world, the term weekday type is also used), which are combined in individual cases with traffic restrictions on the traffic days valid for the journey.

The DINO relation day_type.din enumerates all types of operating days of a data delivery. These are referred to as day types. For example, you could define MonFriS as Monday - Friday, during the school term excluding public holidays and, as a complement to it, MonFriH as Monday - Friday during the school holidays, excluding public holidays. If you wish, you could also define Monday to Fr with public holidays as MoFr, for example.

The day types are linked to a calendar in the day_type_calendar.din relation. Only one day type is assigned to each applicable day within the timetable period. For example, 26 March 2015 could be defined as ThuS (Thursday school) or MonFriS (Monday - Friday school), while in the following Easter holiday week, Holy Thursday, 3 April 2015, for example, could be defined as day type ThuH (Thursday holiday) or MonFriH (Monday - Friday holiday). These are only examples that can be

modified at any time according to the requirements of the timetable and the economic storage of the same.

Day types are therefore defined for individual days. For a more compact representation, the day types relating to the calendar are grouped together. A day type group is formed with the DINO relation day_type_2_day_attribute.din. The day types can be summarized here. Here is an example:

day_type defines the following day types:

```
VERSION;DAY_TYPE_NR;DAY_TYPE_TEXT;STR_DAY_TYPE;
1;1;Sunday and public holiday;SO;
1;2;Saturday;SA;
1;3;school on Fridays only;FrS;
1;4;School Thursdays only;DoS;
1;5;School Wednesdays only;MiS;
1;6;school on Tuesdays only;DiS;
1;7;school on Mondays only;MoS;
1;8;Fridays only;FrF;
1;9;Thursdays holidays only;DoF;
1;10;Wednesdays holidays only;MiF;
1;11;holidays on Tuesdays only;TueF;
1;12;Mondays holidays only;MoF;
```
Now we can define a composite group Monday to Friday school as day_attribute_no = 1, for example, in the day_type_2_day_attribute.din table

```
VERSION;DAY_TYPE_NR;DAY_ATTRIBUTE_NR;
1;7;1;
1;6;1;
1;5;1;
1;4;1;
1;3;1;
…
```
and another composite day type group Monday to Friday holidays as day_attribute_no =2,

```
…
1;12;2;
1;11;2;
1;10;2;
1;9;2;
1;8;2;
1;3;1;
…
```
and a Saturday day-type attribute as day_attribute_no = 3 as follows:

```
1;2;3;
…
```

The day_attribute.din relation describes the composite day type groups in more detail:
```
VERSION;DAY_ATTRIBUTE_NR;DAY_ATTRIBUTE_TEXT;STR_DAY_ATTRIBUTE;

1;1;MonFriS;Monday - Friday School;
1;2;MonFriF;Monday - Friday Holidays;
1;3;Sat;Saturday;
…
```

In addition to the day type groups, service restrictions are mapped as bit fields in the service_restriction.din relation.

Within a trip (trip.din), the system then references the day type groups (DAY_ATTRIBUTE_NR) or the service restrictions (RESTRICTION) directly. The validity of connection linkages (connection.din) can be limited to day type groups (DAY_ATTRIBUTE_NR).

In summary, service days are set up in the following steps:

1. version.din: Define the timetable period
2. day_type.din: Define the day types that were assigned to the calendar days.
3. day_type_calendar.din: Define the company calendar in the relation and assign day types with DAY_TYPE_NR. This may occur several times in the calendar.
4. day_type_2_day_attribute.din: Group the day types into composite day types.
5. service_restriction.din: Restricting day types to certain time periods via a bit field, e.g. only in the first week of May, only in the Easter holidays, etc.
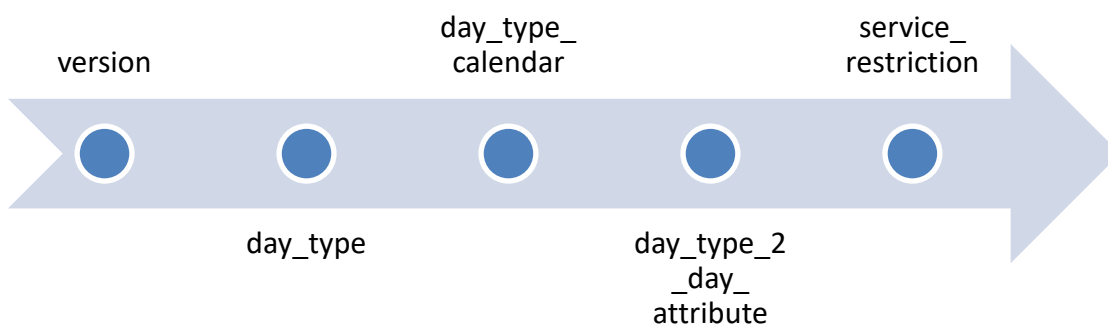


### 5.2.1 version.din

Basic versions, valid standard versions for network, structure and timetable data

This table is required for the DIVA import, it describes the timetable period, the source network and the DINO format used. In principle, it is also possible to export data from several subnetworks of a client system at the same time.

The version is part of the unique identifier of every other DINO table.

| | | | **Table: version** | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | - |
| | N | Optional | VERSION_TEXT | char (70) | | Version description | - |
| | N | Optional | TIMETABLE_PERIOD | char (4) | | Abbreviation of the timetable version | - |
| | N | Optional | TT_PERIOD_NAME | char (40) | | Depending on the input specification settings:<br><br>Network: <Network>, project: <project>, plan <map type/coordinates>, date: <export date> <export time> | - |

| | | Table: version | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Optional | PERIOD_ DATE_FROM | date (YYYYMMDD) | | Date from which the general timetable version applies: e.g. the number 20021231 refers to 31 December 2002 | - |
| | N | Optional | PERIOD_DATE_TO | date (YYYYMMDD) | | Date until which the general timetable version applies: e.g. the number 20021231 refers to 31 December 2002 | - |
| | N | Optional | NET_ID | char (3) | | Provider / network name | - |
| | N | Optional | PERIOD_PRIORITY | decimal (1) | | Weighting of the timetable version | - |
| | N | Obligation (mandatory) | DINO_FORMAT | char (40) | | Mandatory as of DINO 2.2. With older formats, the column may also be missing. | - |

*Table 3 Version - Timetable projects and supplier networks*

The weight of the timetable period(PERIOD_PRIORITY) is to be set if there are several timetables overlapping in time. The weighting then determines which timetable period is to be prioritised on a particular date. The timetable project with the higher weighting is prioritised. Currently not implemented in DINO import, as recorded in DIVA.

The DIVA 4 DINO import uses the date set in version.din(PERIOD_DATE_FROM, PERIOD_DATE_TO) to determine whether an existing line version should be overwritten in DIVA (same date as previous delivery) or whether a new line version must be created.

The validity start date of a line version created by the DIVA 4 DINO import is identical to the date of the import. unless the PERIOD_DATE_FROM field in the version.din table is filled, in which case the validity starting date of the new line version is defined by PERIOD_DATE_FROM.

**Ex:**

```
VERSION;VERSION_TEXT;TIMETABLE_PERIOD;TT_PERIOD_NAME;PERIOD_DATE_FROM;PERIOD_DATE_TO;NET_ID;P
ERIOD_PRIORITY;
1;Timetable period 2013/2014;FP22;Timetable period from 15.12.2013;20131215;20141213;ovb;1;
```

A DINO data delivery is generally to be considered self-contained. The version.din file describes DINO versions, which mostly correspond to DIVA timetable periods. However, it is also possible to create an export with a limited validity period (e.g. Today until 270 days from now). In that case, the validity (PERIOD_DATE_FROM & PERIOD_DATE_TO) does not denote the DIVA schedule period, but the validity period of the record. New line versions are imported into DIVA with corresponding validity.

In the case of a DINO export of several DIVA timetable projects, each project is mapped to a DINO version. If the projects overlap, the one with the higher priority (PERIOD_PRIORITY) on the selected deadline is to be used. A DINO version is self-contained, so the routes of version 2 also refer to the stopping points of version 2 (stop.din, stop_area.din, stop_point.din).

```
VERSION;VERSION_TEXT;TIMETABLE_PERIOD;TT_PERIOD_NAME;PERIOD_DATE_FROM;PERIOD_DATE_TO;NET_ID;P
ERIOD_PRIORITY;
 1;Subnet: kvv, project: j21, Plan: MRCV, Date: 24.06.2021 07:58:27 ;j21 ;Valid from 13.12.20
to 12.06.21 (W) ;20201213;20210612;kvv;1;
```

```
 2;Subnet: kvv, project: s21, Plan: MRCV, Date: 24.06.2021 07:58:43 ;s21 ;Valid from 13.06.21
to 11.12.21 (S) ;20210613;20211211;kvv;1;
```

The consideration is to be extended to the lines. If a DINO line with the same LINE_NR is delivered several times, the one whose version is valid on the respective reference date shall be used. The example shows line 107 (LINE_NR) with the routes 1 - 6 (STR_LINE_VAR), once each for the timetable period j21 which corresponds to DINO version 1 and s21 which corresponds to DINO version 2. Until 12.06 the line 107 from version 1 is to be used, from 13.06 the line with the assigned version 2 is valid.

```
VERSION;BRANCH_NR;LINE_NR;STR_LINE_VAR;LINE_NAME;LINE_DIR_NR;LAST_MODIFIED;MOT_NR;VALID_FROM;
VALID_TO;OP_CODE;OBO_SHORT_NAME;ROUTE_TYPE;GLOBAL_ID;BIKE_RULE;
 1; 6;       1;    1;107                                     ;  1;10.05.2021  14:11:18;
5;20201213;20210612;01 ;AVG ; 0;                                                    ;-1;
 1; 6;       1;    2;107                                     ;  1;10.05.2021  14:11:18;
5;20201213;20210612;01 ;AVG ; 0;                                                    ;-1;
 1; 6;       1;    3;107                                     ;  1;10.05.2021  14:11:18;
5;20201213;20210612;01 ;AVG ; 0;                                                    ;-1;
 1; 6;       1;    4;107                                     ;  2;10.05.2021  14:11:18;
5;20201213;20210612;01 ;AVG ; 0;                                                    ;-1;
 1; 6;       1;    5;107                                     ;  2;10.05.2021  14:11:18;
5;20201213;20210612;01 ;AVG ; 0;                                                    ;-1;
 1; 6;       1;    6;107                                     ;  2;10.05.2021  14:11:18;
5;20201213;20210612;01 ;AVG ; 0;                                                    ;-1;
 2; 6;       1;    1;107                                     ;  1;28.10.2020  14:30:53;
5;20210613;20211211;01 ;AVG ; 0;                                                    ;-1;
 2; 6;       1;    2;107                                     ;  1;28.10.2020  14:30:53;
5;20210613;20211211;01 ;AVG ; 0;                                                    ;-1;
 2; 6;       1;    3;107                                     ;  1;28.10.2020  14:30:53;
5;20210613;20211211;01 ;AVG ; 0;                                                    ;-1;
 2; 6;       1;    4;107                                     ;  2;28.10.2020  14:30:53;
5;20210613;20211211;01 ;AVG ; 0;                                                    ;-1;
 2; 6;       1;    5;107                                     ;  2;28.10.2020  14:30:53;
5;20210613;20211211;01 ;AVG ; 0;                                                    ;-1;
 2; 6;       1;    6;107                                     ;  2;28.10.2020  14:30:53;
5;20210613;20211211;01 ;AVG ; 0;                                                    ;-1;
```

### 5.2.2 day_type_calendar.din

Company or corporate calendar, assigns day types to operating days

This table is required. It maps the day types listed in day_type.din to calendar days.

| Table: day_type_calendar | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|  | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | Version.version |
|  | Y | Obligation (mandatory) | DAY | Date (YYYYMMDD) |  | Date |  |
|  | N | Optional | DAY_TEXT | char(40) |  | Description |  |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| | | **Table: day_type_calendar** | | | | | |
| | N | Obligation (mandatory) | DAY_TYPE_NR | decimal (9) | | Day type number | |

*Table 4 Day_Type_Calendar - Company calendar*

**Ex:**

```
VERSION;DAY;DAY_TEXT;DAY_TYPE_NR

1;"20111121";"";1
1;"20111122";"";2
1;"20111123";"";3
1;"20111124";"";4
1;"20111125";"";5
1;"20111126";"";6
1;"20111127";"";7
1;"20111128";"";1
1;"20111129";"";2
1;"20111130";"";3
```
…

### 5.2.3 day_type.din

Day types - a list of all types of operating days, Mon, Tue, Wed, Thu, Fri, Sat, Sun

This table is required. It contains text descriptions of the day types that are assigned to calendar days in the day_type_calendar.din relation.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| | | **Table: day_type** | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | version.version |
| | Y | Obligation (mandatory) | DAY_TYPE_NR | decimal (9) | | Day type number | |
| | | Optional | DAY_TYPE_TEXT | char (40) | | Description of the day type | |
| | | Optional | STR_DAY_TYPE | char (2) | | Short designation of the day type | |

*Table 5 Day_Type - Day types, operating days / weekdays*

The traffic days, i.e. those days on which vehicles actually operate, are determined from the operating days. These may be, for example, the days of a week:

**Ex:**

```
VERSION;DAY_TYPE_NR;DAY_TYPE_TEXT;STR_DAY_TYPE
1;1; "Monday 21.11.2011"; "Mo"
1;2; "Tuesday 22.11.2011"; "Tue"
1;3; "Wednesday 23.11.2011"; "Wed"
1;4; "Thursday 24.11.2011"; "Do"
1;5; "Friday 25.11.2011"; "Fri
1;6; "Saturday 26.11.2011"; "Sat"
1;7; "Sunday 27.11.2011"; "Su"
1;8; "Monday 28.11.2011"; "Mo"
…
```

### 5.2.4   day_type_2_day_attribute.din

Assignment of day types to day-type attributes (to form groups)

This table is required, it groups the day types mapped on the calendar into day type groups.

| | | Table: day_type_2_day_attributes | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | day_type.version day_attribute.version |
| | Y | Obligation (mandatory) | DAY_TYPE_NR | decimal (9) | | Day type number | day_type.day_type_nr |
| | Y | Obligation (mandatory) | DAY_ATTRIBUTE_NR | decimal (5) | | Day-type attribute number | day_attribute.day_attribute_nr |

*Table 6 Day_Type_2_Day_Attribute - Grouping of day types*

Day types are grouped into day-type attributes in the day_type2_day_attribute.din table.

A text description of the groups can be provided in the day_attribute.din table.

**For example:**

```
VERSION;DAY_TYPE_NR;DAY_ATTRIBUTE_NR
1;1;1
1;2;2
1;3;3
1;4;4
1;5;5
1;6;6
1;7;7
…
```

### 5.2.5   day_attribute.din

Day-type attributes, a group of day types,

This table is required. It contains text descriptions of the grouped day types. These are called day-type attributes.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| | | **Table: day_attributes** | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | version.version |
| | Y | Obligation (mandatory) | DAY_ATTRIBUTE_NR | decimal (5) | | Day-type attribute number | - |
| | N | Obligation (mandatory) | DAY_ATTRIBUTE_TEXT | char (40) | | Identifier | - |
| | N | Optional | STR_DAY_ATTRIBUTES | char (2) | | Short designation, blank | - |

*Table 7 Day_Attribute - Day type attributes, text descriptions*

A combination or group of day types is aggregated in the day_type_2_day_attribute.din table to form a day-type attribute. The day_attribute.din table serves to describe the day-type attribute in greater detail.

**Ex:**

```
VERSION;DAY_TYPE_NR;DAY_TYPE_TEXT;STR_DAY_TYPE
1;1; "Monday 21.11.2011"; "Mo"
1;2; "Tuesday 22.11.2011"; "Tue"
1;3; "Wednesday 23.11.2011"; "Wed"
1;4; "Thursday 24.11.2011"; "Do"
1;5; "Friday 25.11.2011"; "Fri
1;6; "Saturday 26.11.2011"; "Sat"
1;7; "Sunday 27.11.2011"; "Su"
1;8; "Monday 28.11.2011"; "Mo"
```

Note on the DIVA 4 DINO import: Alternatively, DIVA 4 day types can be specified in the import configuration, the evaluation of *DAY_ATTRIBUTE_TEXT* by the DIVA 4 DINO import is omitted. If

this does not happen, the day type specified in *DAY_ATTRIBUTE_TEXT* is evaluated and used by the DIVA 4 DINO import.

### 5.2.6    service_restriction.din

Additional service restriction applicable to a day-type attribute
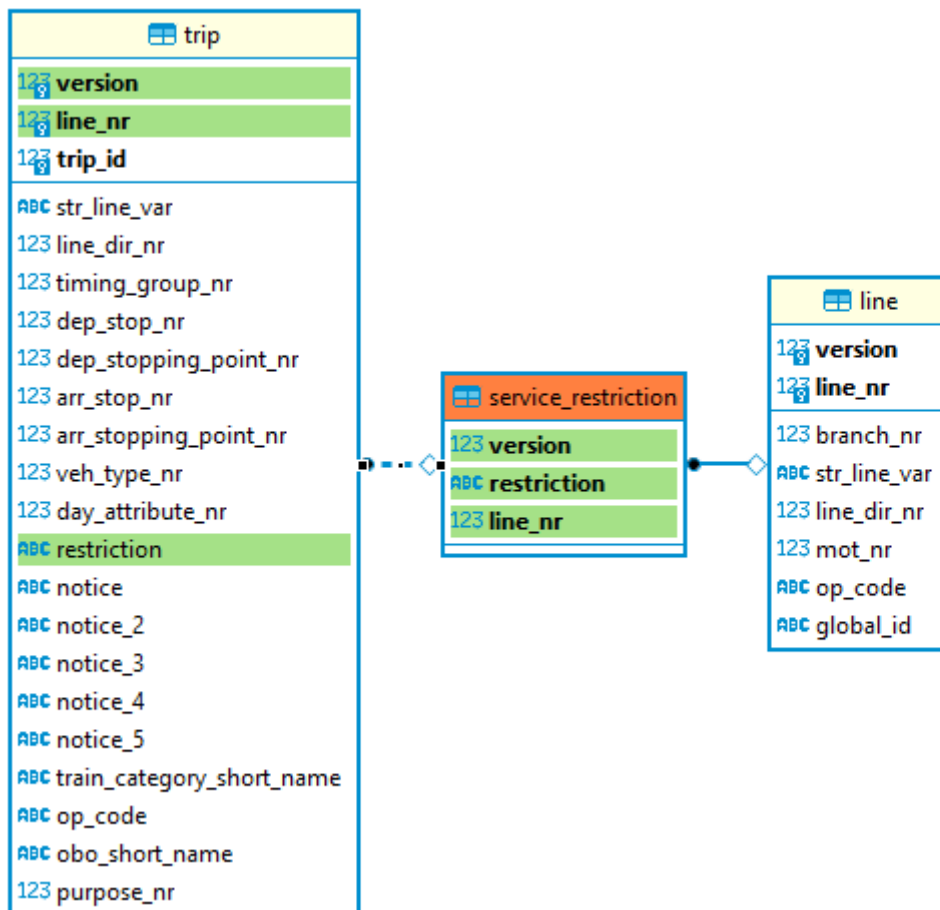


*Figure 3 ER diagram Service_Restriction*

The day types grouped in the day-type attributes can be restricted by service restrictions defined in the service_restriction.din table (e.g. "from 02.05 to 13.06", "only in the school holidays", etc.). The days set in the bit field are the days on which the service will run.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | Foreign Key to |
|---|---|---|---|---|---|---|---|
| **Table: service_restriction** | | | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | line.version |
| | Y | Obligation (mandatory) | RESTRICTION | char (10) | | Traffic restriction in DIVA up to 10-points.  Change in version 2.3: Extension from char(5) to char(10) | |
| | N | Optional | RESTRICT_TEXT1 | char (60) | | Text row 1 | |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | Foreign Key to |
|---|---|---|---|---|---|---|---|
| | N | Optional | RESTRICT_TEXT2 | char (60) | | Text row 2 | |
| | N | Optional | RESTRICT_TEXT3 | char (60) | | Text row 3 | |
| | N | Optional | RESTRICT_TEXT4 | char (60) | | Text row 4 | |
| | N | Optional | RESTRICT_TEXT5 | char (60) | | Text row 5 | |
| | N | Obligation (mandatory) | RESTRICTION_DAYS | char (192) | | Binary code of the service days | |
| | N | Obligation (mandatory) | DATE_FROM | Date (YYYYMMDD) | | Start date | |
| | N | Obligation (mandatory) | DATE_UNTIL | Date (YYYYMMDD) | | End date | |
| | Y | Optional | LINE_NR | decimal (8) | | Internal line number | line.line_nr together with version |

**Table: service_restriction**

*Table 8 Service_Restriction -Traffic Restrictions on Day Type Characteristics*

The Restriction_Text, if supplied, is only evaluated if the VB is also used.

**Generation or interpretation of the binary code:**

The service restrictions (SR) are coded for the period of the exported timetable period, i.e. monthly, in a

DWORD = long integer = 4 bytes = 32 bits.

The LSB (least significant bit, bit 0) corresponds to the first of the month.
The MSB (most significant bit, bit 31) remains free.

A 12-month period therefore results in 12 DWORDs. If they are output in hexadecimal notation, this results in eight characters per DWORD.

A SR that is valid for 12 months can then be represented as a 12*8 = 96 character string.
DIVA can have a maximum duration of two years but it is rarely used for more than one year.

The theoretical maximum length of the string is therefore 24 * 8 = 192 characters.

The period for which the validity of the SR is defined is limited by the DATE_FROM and DATE_UNTIL fields.
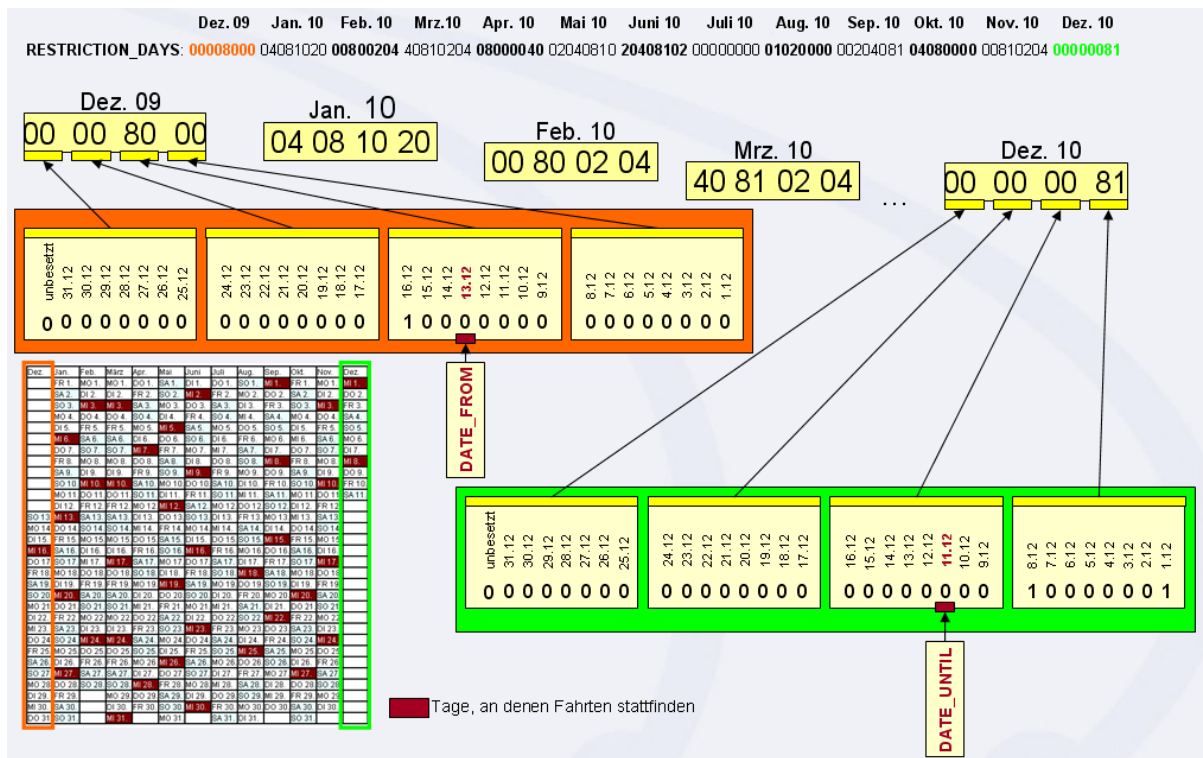
*Figure 4 Graphical view of the binary code*

**Ex:**

```
VERSION;RESTRICTION;RESTRICT_TEXT1;RESTRICT_TEXT2;RESTRICT_TEXT3;RESTRICT_TEXT4;RESTRICT_TEXT
5;RESTRICTION_DAYS;DATE_FROM;DATE_UNTIL;

1;8;;;;;;7FC000000000003F003F000000000000003FF80010000001000403C07FFFFFF07FFFFFFF0000003F0200
00000000000100000080;20131215;20141213;

1;31;;;;;;7FC0000000000027000000000000000000003FE0001000000100040380 7FFFFFC07FFFFFFF0000001F020
00000000000100000080;20131215;20141213;

1;34;;;;;;7FC000000000002F000000000000007C03FFE00010000001001FFF80600000007FFFFFFF00007FFF7E0
0000400000000100000080;20131215;20141213;
```

## 5.3   Location data

Location data includes all stop data. The modelling of stops in DINO format is strongly based on the DIVA data model. The model is hierarchical. Stops can have 1..n areas, each area can have 0..n platforms. Stops that have no areas can have 0..n risers. The platforms are modelled in a virtual area with the number "0".
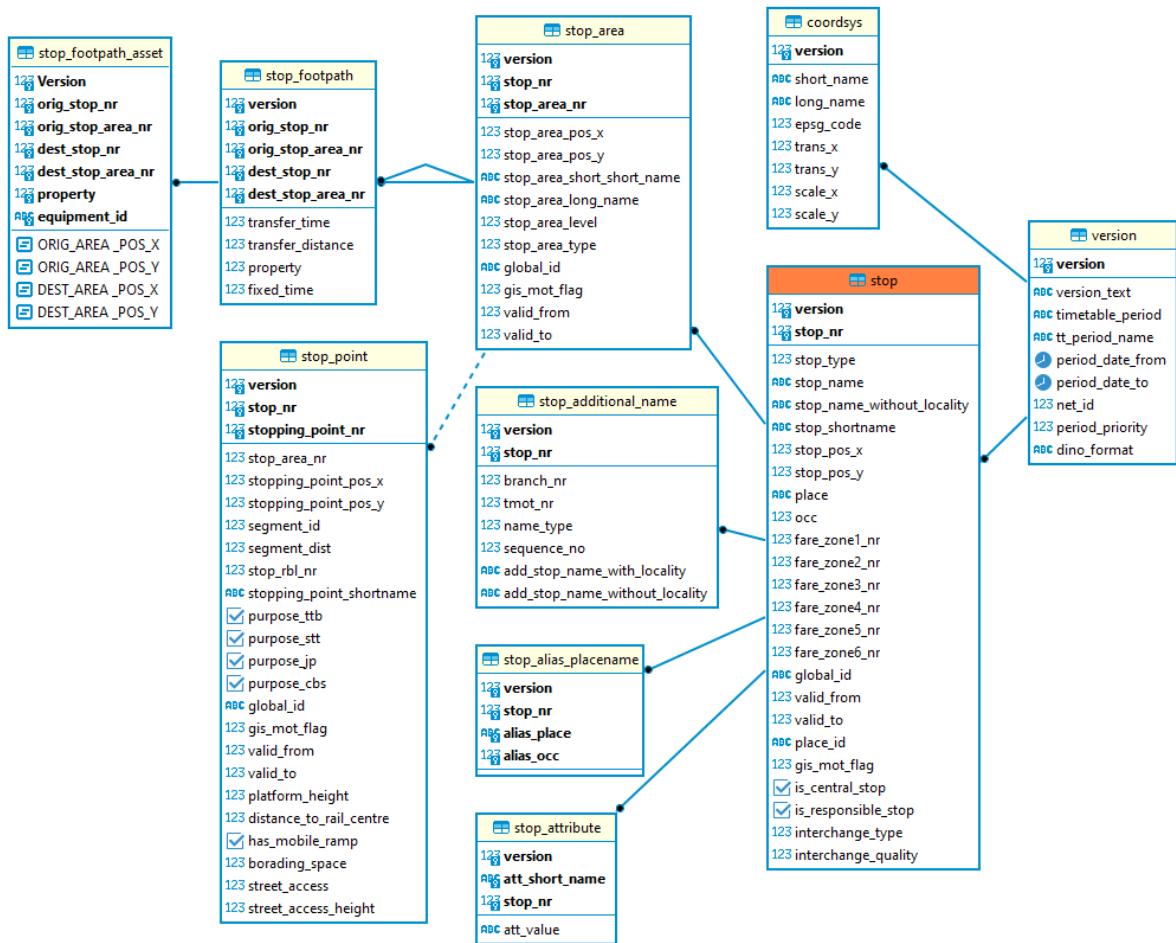
*Figure 5 ER diagram location data*

### 5.3.1 stop.din

Stops

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments |
|---|-----|-----------------|----------------|-----------|-------------|----------|
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | For stops: stop number |

| | | Table: stop | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** |
| | N | Optional | STOP_TYPE | decimal (2) | 0-99 | DIVA 4 stop type<br><br>0 = Normal stop,<br><br>1 = Demand stop in the network area,<br><br>2 = For disembarking only,<br><br>3 = Request stop,<br><br>4 = Demand stop not in network area,<br><br>6 = Transition fare<br><br>8 = Inbound and outbound trips,<br><br>9 = Not in network area,<br><br>10 = Time position,<br><br>12 = School stop |
| | N | Obligation (mandatory) | STOP_NAME | char (255) | | Stop name (with locality)<br><br>Expanded from 50 to 255 digits in version 2.3. |
| | N | Optional | STOP_NAME_WITHOUT_LOCALITY | char (255) | | Stop name without locality<br><br>Expanded from 50 to 255 digits in version 2.3. |
| | N | Optional | STOP_SHORTNAME | char (8) | | Abbreviations |
| | N | Optional | STOP_POS_X | decimal (12) | | X coordinate (e.g. WGS 84)<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>-1 or blank entry means no coordinate |
| | N | Optional | STOP_POS_Y | decimal (12) | | Y- coordinate<br><br>If WGS84, then a value with up to 7 decimal places<br><br>48.1234567<br><br>-1 or blank entry means: no coordinate |
| | N | Optional | PLACE (deprecated) | char (20) | | Name of the locality or district in which the stop is located (max. 20 characters) (obsolete, to be replaced by PLACE_ID in future) |
| | N | Optional | OCC | decimal (8) | | Official municipal code of the locality |
| | N | Optional | FARE_ZONE1_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set |
| | N | Optional | FARE_ZONE2_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set |

| **Table: stop** | | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** |
| | N | Optional | FARE_ZONE3_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set |
| | N | Optional | FARE_ZONE4_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set |
| | N | Optional | FARE_ZONE5_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set |
| | N | Optional | FARE_ZONE6_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set |
| I F | N | Optional | GLOBAL_ID | Char (128) | ISO 8859-1 | Global stop ID according to IFOPT standard. Field length increased from Char(50) to Char(128) from version 2.2. |
| | N | Optional | VALID_FROM | Decimal (8) | YYYYMMDD | Validity of the stop, to be used as a keyless attribute |
| | N | Optional | VALID_TO | Decimal (8) | YYYYMMDD | Validity of the stop |
| | N | Optional | PLACE_ID | Char (50) | | Will be filled with a global location ID in the future, making the PLACE field obsolete. For now, the district number can be entered here |
| | N | Optional | GIS_MOT_FLAG | Decimal(10) | 0 .. 4294967295 | GIS means of transport flags. This attribute defines the GIS edges that are used for automatic routing. BIT1 = 1 = Pedestrians BIT2 = 2 = Cyclists BIT3 = 4 = Normal private transport BIT4 = 8 = Priority private transport, busses BIT5 = 16 = Rail, standard gauge BIT6 = 32 = Tramway, rail narrow gauge BIT7 = 64 = Subway BIT8 = 128 = Ships, ferries, everything on waterways ... |
| | N | Optional | IS_CENTRAL_STOP | Boolean | (0 \| 1) | Central stop flag |
| | N | Optional | IS_RESPONSIBLE_STOP | Boolean | (0 \| 1) | Indicates that the exporting subnetwork is responsible for this stop. |
| | N | Optional | INTERCHANGE_TYPE | Decimal(1) | 0..2 | 0 = N = never 1 = Y = always 2 = determine automatically |
| | N | Optional | INTERCHANGE_QUALITY | Decimal(2) | 0..99 | Interchange quality (higher = better) Transfer quality |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments |
|---|---|---|---|---|---|---|
| | | | **Table: stop** | | | |
| | N | Optional | FARE_PROVIDER_CODES | Char(255) | | Comma-separated list of tariff knowledge provider codes according to DELFI |

*Table 9 Stop - Stops*

| Attribute name | Foreign key |
|---|---|
| VERSION | fare_zone.version,version.version |
| FARE_ZONE1_NR | fare_zone. Fare_zone1_nr |
| FARE_ZONE2_NR | fare_zone. Fare_zone2_nr |
| FARE_ZONE3_NR | fare_zone. Fare_zone3_nr |
| FARE_ZONE4_NR | fare_zone. Fare_zone4_nr |
| FARE_ZONE5_NR | fare_zone. Fare_zone5_nr |
| FARE_ZONE6_NR | fare_zone. Fare_zone6_nr |

*Table 10 Stop - foreign key*

## STOP_NAME and STOP_NAME_WITHOUT_LOCALITY

For compatibility reasons with older DINO versions, the DINO field *STOP_NAME* is imported to DIVA in the field "Stop name without location", the location addition in the field *PLACE* is not evaluated.

If a stop name is also supplied in the *STOP_NAME_WITHOUT_LOCALITY* field, this name is imported into the DIVA data field "Stop name without location", the location suffix from the DINO data field *PLACE* is evaluated and imported together with the supplied name into the DIVA data field "Stop name with location".

## GLOBAL_ID

The GLOBAL_ID is a unique ID for identifying stop objects. If this optional field is filled, the provider must ensure that it is unique.

## VALID_FROM - VALID_TO

The validity of the stop can optionally be specified here. But be careful when using it in the DINO data model, each location must exist only once. The name at the time of the export applies. Stops are identified by means of the global ID and assigned by the importing systems to their own stop inventories. The importing systems can then use their own stop names or adapt their own if necessary.

If the importing system does not use any assignments and takes over the exported DINO data 1:1, then the importing system receives the names of the exporting system that are valid at the time of the export.

### 5.3.2 stop_area.din

Transfer areas at stops

This table is only required if there are ranges not equal to 0. A DINO data set must then contain this table if at least one exported stop has at least one created area (≠ makeshift area 0).

| Table: stop_area | | | | | | |
|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | Internal stop number |
| | Y | Obligation (mandatory) | STOP_AREA_NR | decimal (5) | 0..99998 | Number of a stop area within a stop |
| | N | Optional | STOP_AREA_POS_X | decimal (12) | XXX.XXXXX | X coordinate (e.g. WGS 84) If WGS84, then a value with up to 7 decimal places 16.1234567 -1 or blank entry means no coordinate |
| | N | Optional | STOP_AREA_POS_Y | decimal (12) | XXX.XXXXX | X coordinate (e.g. WGS 84) If WGS84, then a value with up to 7 decimal places 48.1234567 -1 or blank entry means no coordinate |
| | N | Optional | STOP_AREA_SHORT_NAME | char (5) | | Short designation of a stop area within a stop |
| | N | Optional | STOP_AREA_LONG_NAME | char (20) | | Designation of a stop area within a stop |

| | T | Key | Mandatory field | Attribute name | Data type | Value range | Comments |
|---|---|---|---|---|---|---|---|
| **Table: stop_area** | | | | | | | |
| | | N | Optional | STOP_AREA_LEVEL | decimal (3) | | Nivea |
| | | N | Optional | STOP_AREA_TYPE | decimal (2) | | Area type<br><br>0 .. Access and public transport<br><br>1 .. Public transport only<br><br>2 .. P&R<br><br>3 .. B&R<br><br>4 .. Taxis<br><br>5 .. Access<br><br>6 .. Airport terminal<br><br>7 .. Access and B&R<br><br>8 .. Access, public transport and B&R<br><br>9 .. Access and taxi<br><br>10 .. Access, public transport and taxi<br><br>11 .. Mezzanine level<br><br>12 .. Request stop |
| | IF | N | Optional | GLOBAL_ID | Char (128) | ISO 8859-1 | Global stop ID according to IFOPT standard. Field length increased from Char(50) to Char(128) from version 2.2. |
| | | N | Optional | GIS_MOT_FLAG | Decimal(10) x | 0 .. 4294967295 | GIS means of transport flags. This attribute defines the GIS edges that are used for automatic routing.<br><br>BIT1 = 1 = Pedestrians<br><br>BIT2 = 2 = Cyclists<br><br>BIT3 = 4 = Normal private transport<br><br>BIT4 = 8 = Priority private transport, busses<br><br>BIT5 = 16 = Rail, standard gauge<br><br>BIT6 = 32 = Tramway, rail narrow gauge<br><br>BIT7 = 64 = Subway<br><br>BIT8 = 128 = Ships, ferries, all water transport |
| | | N | Optional | VALID_FROM | Decimal(8) | YYYYMMDD | Validity of the transfer area |

| | Table: stop_area | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** |
| | N | Optional | VALID_TO | Decimal(8) | YYYYMMDD | Validity of the transfer area |

*Table 11 Stop_Area - Stop areas*

### 5.3.3    stop_point.din

Stopping points

This table is required.

The table describes the stops or platforms at a stop. The Stopping_Point_No is incremented over the entire stop, regardless of the individual areas.

| | Table: stop_point | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | stop_area.version |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | Internal stop number | stop_area.stop_nr |
| | N | Obligation (mandatory) | STOP_AREA_NR | decimal (5) | 0..99998 | Number of a stop within a locality | stop_area.stop_area_nr (except generic areas Stop_Area_No "0") |
| | Y | Obligation (mandatory) | STOPPING_POINT_NR | decimal (2) | 0..99 | Stopping point number | - |
| | N | Optional | STOPPING_POINT_POS_X | decimal (12) | XXX.XXXXX | Stopping point coordinate X<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>-1 or blank entry means no coordinate | - |
| | N | Optional | STOPPING_POINT_POS_Y | decimal (12) | XXX.XXXXX | Stopping point coordinate Y<br><br>If WGS84, then a value with up to 7 decimal places<br><br>48.1234567<br><br>-1 or blank entry means no coordinate | - |
| | N | Optional | SEGMENT_ID | decimal (10) | | GIS segment ID | - |
| | N | Optional | SEGMENT_DIST | decimal (8) | | Distance from the first node | - |
| | N | Optional | STOP_RBL_NR | decimal (7) | | AVL stop number (according to VDV-454) | - |
| | N | Optional | STOPPING_POINT_SHORTNAME | char (255) | | DIVA public platform number | - |

| Table: stop_point | | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Optional | PURPOSE_TTB | BOOL | 0, 1 | Used for typesetting | - |
| | N | Optional | PURPOSE_STT | BOOL | 0, 1 | Used for AHF, | - |
| | N | Optional | PURPOSE_JP | BOOL | 0, 1 | Used for EFA, | - |
| | N | Optional | PURPOSE_CBS | BOOL | 0, 1 | Used for ZOB, | - |
| I F | N | Optional | GLOBAL_ID | Char (128) | ISO 8859-1 | Global stop ID according to IFOPT standard. Field length increased from Char(50) to Char(128) from version 2.2. | - |
| | N | Optional | GIS_MOT_FLAG | decimal (10) x | 0 .. 42949672 95 | GIS means of transport flags. This attribute defines the GIS edges that are used for automatic routing.<br><br>BIT1 = 1 = Pedestrians<br><br>BIT2 = 2 = Cyclists<br><br>BIT3 = 4 = Normal private transport<br><br>BIT4 = 8 = Priority private transport, busses<br><br>BIT5 = 16 = Rail, standard gauge<br><br>BIT6 = 32 = Tramway, rail narrow gauge<br><br>BIT7 = 64 = Subway<br><br>BIT8 = 128 = Ships, ferries, all water transport | - |
| | N | Optional | VALID_FROM | decimal (8) | YYYYMM DD | Validity of the stopping point | - |
| | N | Optional | VALID_TO | decimal (8) | YYYYMM DD | Validity of the stopping point | - |
| | N | Optional | PLATFORM_HEIGH T | decimal (4) | | Height of platform in mm above top of rail or street surface | - |
| | N | Optional | DISTANCE_TO_RAI L_CENTRE | decimal (4) | | Horizontal distance of platform edge to centre of rail. Used for calculating gap between vehicle and rail edge in mm | - |
| | N | Optional | HAS_MOBILE_RAM P | Boolean | (0 \| 1) | Platform has a mobile ramp: 0 = No, 1 = Yes | - |
| | N | Optional | BOARDING_SPACE | decimal (4) | | Space (mm) provided on pavement for operating the equipment | - |

| Table: stop_point | | | | | | |
|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | N | Optional | STREET_ACCESS | decimal (1) | | Accessibility of platform from street: 0 = Unknown, 1 = Level(no steps), 2 = Small step, 3 = Large step, 4 = Specified height in STREET_ACCESS_HEIGHT | - |
| | N | Optional | STREET_ACCESS_ HEIGHT | decimal (4) | | Specified height (mm) for steps | - |

*Table 12 Stop_Point - Breakpoints*

PURPOSE_CBS is currently not evaluated by the DIVA 4 DINO import.

### 5.3.4 stop_footpath

Footpaths

Optional table.

| Table: stop_footpath | | | | | | |
|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | stop_area.version |
| | Y | Obligation (mandatory) | ORIG_STOP_NR | decimal (5) | 1..99999 | Internal stop number of the start stop | stop_area. stop_nr |
| | Y | Obligation (mandatory) | ORIG_STOP_AREA_NR | decimal (5) | 1..99998 | internal number start stop area | stop_area. stop_area_nr |
| | Y | Obligation (mandatory) | DEST_STOP_NR | decimal (5) | 1..99999 | Internal stop number of the destination stop | stop_area. STOP_NR |
| | Y | Obligation (mandatory) | DEST_STOP_AREA_NR | decimal (5) | 1...99998 | internal number destination stop area | stop.area. STOP_AREA_NR |
| | N | Obligation (mandatory) | TRANSFER_TIME | decimal (5) | 0..99999 | Transfer time in seconds | - |
| | N | Optional | TRANSFER_DISTANCE | decimal (5) | 0..99999 | Transition path in metres, < -1 = footpath closed, -2 = footpath automatically closed | - |

| | Table: stop_footpath | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Optional | PROPERTY | decimal (2) | 0..99 | Footpath properties<br><br>0 .. Overview as in DivaWeb (can be optionally created via parameters during export, ignored during import into DIVA)<br><br>1 .. Level (default)<br><br>2 .. Stairs<br><br>3 .. Escalator<br><br>4 .. Elevator, lift<br><br>5 .. Ramp<br><br>6 .. Unlit footpath<br><br>7 .. Footpath blocked<br><br>8 .. Without attribution | - |
| | N | Optional | FIXED_TIME | decimal (5) | 0..99999 | Time stipulated by the data provider (may differ from the calculated time for composite footpaths). | - |

*Table 13 Stop_Footpath - Optional exported footpaths*

The DIVA data model basically defines footpaths between areas of stops.

A mapping of a walk from an area to itself is also possible, the transfer time (TRANSFER_TIME) is then interpreted in the DIVA model by the EFA router as a transfer time between means of transport. In this case, the length of the walk (TRANSFER_DISTANCE) must be specified as 0.

### 5.3.5 stop_footpath_asset

Additional stop names

Optional table.

This optional table contains the connecting objects used for a footpath, such as lifts, ramps, escalators and fixed stairs. Export and import of this table are only possible if the DIVA add-on module "Asset Management" (management of stop equipment) is installed. One or more such objects can be assigned to each footpath from DIVA. The detailed information on the referenced objects can be found in the DELFI-csv interface. See also the document "DELFI_CSV_Schnittstelle_V1.12.pdf" or subsequent versions thereof.

| | Table: stop_footpath_asset | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | stop_footpath.VERSION |

| | | Table: stop_footpath_asset | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | ORIG_STOP_NR | decimal (5) | 1..99999 | Internal stop number of the start stop | stop_footpath.ORIG_STOP_NR |
| | Y | Obligation (mandatory) | ORIG_STOP_AREA_NR | decimal (5) | 1..99998 | internal number start stop area | stop_footpath.ORIG_STOP_AREA_NR |
| | Y | Obligation (mandatory) | DEST_STOP_NR | decimal (5) | 1..99999 | Internal stop number of the destination stop | stop_footpath.DEST_STOP_NR |
| | Y | Obligation (mandatory) | DEST_STOP_AREA_NR | decimal (5) | 1...99998 | internal number destination stop area | stop_footpath.DEST_STOP_AREA_NR |
| | Y | Obligation (mandatory) | PROPERTY | decimal (2) | 0..99 | Footpath properties<br><br>1 .. Level (default)<br><br>2 .. Stairs<br><br>3 .. Escalator<br><br>4 .. Elevator, lift<br><br>5 .. Ramp<br><br>6 .. Unlit footpath | - |
| | Y | Obligation (mandatory) | EQUIPMENT_ID | char(100) | | Reference to a connection object from the DELFI csv files<br><br>DELFI_STAIR.csv<br><br>DELFI_Escalator.csv<br><br>DELFI_Elevator.csv<br><br>DELFI_Ramp.csv | - |
| | N | Optional | ORIG_AREA _POS_X | decimal (12) | XXX.XXXXX | X-coordinate of the object on the start stop area<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>-1 or blank entry means no coordinate | - |
| | N | Optional | ORIG_AREA _POS_Y | decimal (12) | XXX.XXXXX | Y-coordinate of the object on the start stop area<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>-1 or blank entry means no coordinate | - |

| | Table: stop_footpath_asset | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Optional | DEST_AREA _POS_X | decimal (12) | XXX.XXXX X | X-coordinate of the object on the target stop area<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>-1 or blank entry means no coordinate | - |
| | N | Optional | DEST_AREA _POS_Y | decimal (12) | XXX.XXXX X | Y-coordinate of the object on the target stop area<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>-1 or blank entry means no coordinate | - |

*Table 14 Stop_Footpath_Asset - Optional data on connection elements*

The ID of the referenced connection objects is assigned in the DIVA system and must of course match the IDs in the DELFI.csv files. No further assumptions should be made about the IDs. It is likely that the ID assigned by DIVA will be replaced by a global infrastructure ID in future expansion stages. This should be possible without changing the format.

The coordinates of the connection objects per DIVA area are optional. They are intended for "sloping" connecting objects such as stairs or ramps that can have different coordinates per level. If such data is not available, the fields should be left empty - the centre point coordinate of the connection objects can already be found in the DELFI-csv files.

### 5.3.6 stop_additional_name.din

Additional stop names

Optional table.

| | Table: stop_additional_name | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | means_of_transport_desc .version,<br><br>stop.version,<br><br>branch.version |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | Original internal stop number | Stop.stop_No |

| Table: stop_additional_name | | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Optional | BRANCH_NR | decimal (2) | 0..99 | Designation of the operating branch or authority | Branch.Branch_No |
| | Y | Optional | TMOT_NR | decimal (2) | | Number of the connecting means of transport | means_of_transport_desc .tmot_nr |

# DINO 2.3

| | Y | Optional | NAME_TYPE | decimal (2) | | Type | - |
|---|---|---|---|---|---|---|---|
| | | | | | | -1 = Standard name (GER: General stop name), | |
| | | | | | | 0 = ON Used when printed as connection stop (GER: connection), | |
| | | | | | | 1 = EF Used for journey planner (GER: EFA), | |
| | | | | | | 2 = ZN Used as additional name (GER: Additional name), | |
| | | | | | | 3 = AH Used for Stop timetable (GER: Posting timetable), | |
| | | | | | | 4 = EZ Used for Final destination (GER: Final destination | |
| | | | | | | 5 = HV Used for Stop index (GER: List of stops | |
| | | | | | | 6 = HZ Used for Additionally in stop index (GER: Additionally in the stop directory | |
| | | | | | | 7 = SA Used for City map section (GER: City map extract), | |
| | | | | | | 8 = SK Used for Type area code (GER: Sentence abbreviation), | |
| | | | | | | 9 = SP Used for Automated timetable service (GER: Voice output), | |
| | | | | | | 10 = ST Used for Stop point lettering (GER: Climb inscription), | |
| | | | | | | 11 = VL Used for Route network map (GER: traffic route plan), | |
| | | | | | | 12 = VR Used for Network map (GER: Composite Area Plan), | |
| | | | | | | 13 = ZO Used for Central stop timetable (GER: ZOB exit), | |
| | | | | | | 14 = Used for FareMatrixShortName (GER: Tariff matrix Hst short name), | |
| | | | | | | 15 = Used for FareMatrixName (GER: Tariff Matrix Destination Name), | |
| | | | | | | 16 = Used for FareMatrixPrintName (GER: Hst print name), | |
| | | | | | | 17 = Used for AnnouncementText (GER: Announcement text), | |
| | | | | | | 18 = Used for OnBoardComputers (GER: on-board computer), | |

| | | Table: stop_additional_name | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | | | | | | 19 = Used for DPI texts (GER: passenger information),<br><br>20 = Operational Print (GER: Operational print output)<br><br>21 = Used for DELFI stop name (GER: DELFI stop name) | |
| | Y | Optional | SEQUENCE_NO | decimal (2) | | Sequence number for the same type and means of transport/branch of operation | - |
| | N | Obligation (mandatory) | ADD_STOP_NAME_WITH_LOCALITY | char (255) | | | - |
| | N | Optional | ADD_STOP_NAME_WITHOUT_LOCALITY | char (255) | | | - |

*Table 15 Stop_Additional_Name - Optional additional stop names*

**keys:**

The attributes (VERSION, STOP_NR, BRANCH_NR, TMOT_NR, NAME_TYPE, SEQUENCE_NO) form the key. Individual values in the columns BRANCH_NR, TMOT_NR, NAME_TYPE, SEQUENCE_NO can also be empty. However, one value of these columns must always be occupied.

**Ex:**

| VERSION | STOP_NR | BRANCH_ | TMOT_NR | NAME_TY | SEQUENC | ADD_STOP_NAME_WITH_LOCALITY | ADD_STOP_NAME_WITHOUT_LOCALITY |
|---|---|---|---|---|---|---|---|
| 1 | 100 | 1 | | | | Hbf (Gl. 5-36) | Hbf (Gl. 5-36) |
| 1 | 100 | 2 | | | | München Hbf | München Hbf |
| 1 | 100 | 3 | | | | München Hbf | München Hbf |
| 1 | 100 | 4 | | | | München Hbf | München Hbf |
| 1 | 100 | 15 | | | | München Hbf | Hauptbahnhof |
| 1 | 100 | 5 | | | | München Hbf | München Hbf |
| 1 | 100 | 6 | | | | München Hbf | München Hbf |
| 1 | 100 | | | | 0 | München Hbf | Hauptbahnhof |
| 1 | 100 | | | | 5 | Hauptbahnhof (Gl. 5-36) | Hauptbahnhof (Gl. 5-36) |
| 1 | 100 | | | | 7 | Hbf (Gl. 5-36) | Hbf (Gl. 5-36) |
| 1 | 100 | | | | 9 | München Hauptbahnhof Gleis 5 bis 36 | Hauptbahnhof Gleis 5 bis 36 |
| 1 | 100 | | | | 11 | Hauptbahnhof (Gl. 5-36) | Hauptbahnhof (Gl. 5-36) |
| 1 | 100 | | | | 12 | Hauptbahnhof (Gl. 5-36) | Hauptbahnhof (Gl. 5-36) |
| 1 | 100 | | | | -1 | München Hbf (Gl. 5-36) | Hauptbahnhof (Gl. 5-36) |

### 5.3.7   stop_alias_placename.din

Aliases for place names

Optional table.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | stop.version |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | Internal stop number | stop.stop_nr |
| | Y | Obligation (mandatory) | ALIAS_PLACE | char (20) | | Name of the locality or district where the stop is located (max. 20 characters) | - |
| | Y | Obligation (mandatory) | ALIAS_OCC | decimal (8) | | Official municipal code of the locality | - |

*Table 16 Stop_Alias_Placename - Optional, additional place names, place parts*

**Ex:**

```
VERSION;STOP_NR;ALIAS_PLACE;ALIAS_OCC;
 1; 3212; Sindelfingen ;08115045;
 1; 6001; Vaihingen (Stgt.)   ;08111000;
 1; 6002; Vaihingen (Stgt.)   ;08111000;
```

### 5.3.8   coordsys.din

Coordinate system

Optional table.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | version.version |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| **Table: coordsys** | | | | | | | |
| | Y | Optional | SHORT_NAME | char (15) | e.g. MRCV, WGS84, WG10 | Short name of the DIVA plan format | - |
| | N | Optional | LONG_NAME | char (255) | | Long name of the coordinate system | - |
| | N | Optional | EPSG_CODE | decimal (10) | | EPSG code | - |
| | N | Optional | TRANS_X | decimal (10) | | X translation | - |
| | N | Optional | TRANS_Y | decimal (10) | | Y translation | - |
| | N | Optional | SCALE_X | decimal (10,5) | | Scale X | - |
| | N | Optional | SCALE_Y | decimal (10,5) | | Scale Y | - |

*Table 17 Coordsys - Optional definition of a coordinate system*

The coordsys.din table is used to describe the coordinate systems in which the stopping points are stored. This involves, as is usual in geoinformatics, specifying the EPSG code as well as the translation and scale parameters of the coordinate system used.

Before exchanging coordinates, always check with MENTZ GmbH if the format used is supported.

If the plan format is occupied, the EPSG code is not evaluated by the DIVA import. If the EPSG code is occupied and the SHORT_NAME is not, then coordination with MENTZ GmbH Support is necessary in any case before the import.

## 5.4  Fare data

In the fare data tables, the fare zones as well as the fare transitions between fare zones are modelled.
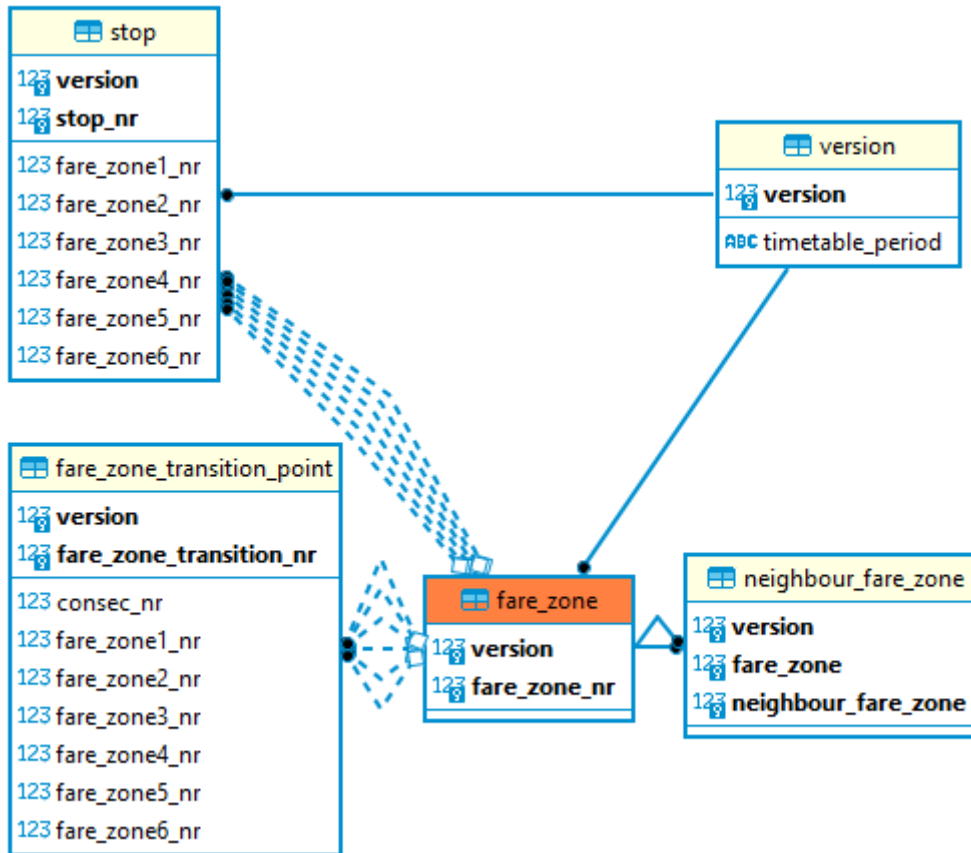
*Figure 6 ER diagram tariff data*

### 5.4.1    fare_zone.din

Fare zones

Optional table.

Analogous to the DIVA table FARE_ZONE:

> *The area covered by a common fare system can be subdivided into different fare zones. The fare is determined by counting the fare zones travelled through (GER: PKM tariff zones).*

| | | | Table: fare_zone | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | version.version |
| | Y | Obligation (mandatory) | FARE_ZONE_NR | decimal (5) | | Fare zone, tariff zone, | - |
| | N | Optional | FARE_ZONE_LONG_NAME | char (50) | | Fare zone, long name | - |
| | N | Optional | FARE_ZONE_TYPE | decimal(1) | | Fare zone type, 0 = normal fare zone, 1 = neutral zone | - |

| | Table: fare_zone | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | N | Optional | FARE_ZONE_COLOR | decimal(18) | | Colour for visualization in DIVA, RGB, 3x8-bit | - |

*Table 18 Fare_Zone - Optional specification of fare zones*

### 5.4.2 neighbour_fare_zone.din

Neighbouring fare zones

Optional table.

| | Table: neighbour_fare_zone | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Datatype | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | fare_zone.version |
| | Y | Obligation (mandatory) | FARE_ZONE | decimal (5) | | Fare zone, tariff zone, | fare_zone.fare_zone_nr |
| | Y x | Obligation (mandatory) | NEIGHBOUR_FARE_ZONE | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set | fare_zone.neighbour_fare _zone |

*Table 19 Neighbour_Fare_Zone - Optional neighbourhood relations of fare zones*

The NEIGHBOUR_FARE_ZONE is part of the key because there is usually more than one neighbouring fare zone.

### 5.4.3 fare_zone_transition.din

Transitions between tariff zones

Optional table.

Analogous to the DIVA table FZ_TRANSITION:

*Fare zone transition records are used to determine the sequence of fare zones if a route skips zones without stopping (Otherwise the sequence can be determined by the stops used) (GER: DIVA3 Table Tariff Zone Transitions)*

| | Table: fare_zone_transition | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | stop.version, branch.version, line.version, fare_zone_transition_point .version |

| | Table: fare_zone_transition | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandator y field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | FARE_ZONE_TRA NSITION_NR | Decimal (18) | | Unique internal non-persistent ID. Serves only to form a reference to fare_zone_transition_point.din | fare_zone_transition_point .FARE_ZONE_TRANSITI ON_NR |
| | N | Obligation (mandatory) | START_STOP_NR | decimal (5) | 1..99999 | First stop number | stop.start_stop_nr |
| | N | Obligation (mandatory) | END_STOP_NR | decimal (5) | 1..99999 | Last stop number | stop.end_stop_nr |
| | N | Obligation (mandatory) | USE_ALWAYS | Boolean | 0..1 | | |
| | N | Obligation (mandatory) | FARE_POINT | Boolean | 0..1 | Transition is a fare point, 0 = no, 1 = yes | |
| | N | Optional | BRANCH_NR | decimal (2) | 0..99 | Foreign key of the operating branch. If set, then valid for all lines of the operating branch. | branch.branch_nr, line.branch_nr |
| | N | Optional | LINE_NR | decimal (8) | | Foreign key of the line. If set, then valid only for this line | line.line_nr |
| | N | Optional | LINE_DIR_NR | decimal (3) | | Foreign key of the direction, must be set together with LINE_NR. If set, it is valid only for line in specified direction | line.line_dir_nr |

*Table 20 Fare_Zone_Transition - Optional table for fare zone transitions between two stops*

Tariff zone transitions are defined in order to define a sequence of tariff zones even if there are zones on the route that have no link to a stop.

The fare_zone_transition.din relation refers to transition points defined along the most direct connection between two stopping points through a fixed sequence and the distance to the previous transition point.

**Ex:**
```
VERSION;FARE_ZONE_TRANSITION_NR;START_STOP_NR;END_STOP_NR;USE_ALWAYS;FARE_
POINT;BRANCH_NR;LINE_NR;LINE_DIR_NR;
 1;    0001;   22;   313;0;0;  ;          ;   ;
 1;    0002;   70;   855;0;0;  ;          ;   ;
 1;    0003;   71;   378;0;0;  ;          ;   ;
 1;    0004;   71;   856;0;0;  ;          ;   ;
 1;    0005;   76;   265;0;0;  ;          ;   ;
…
```

### 5.4.4 fare_zone_transition_point.din

Transition points between tariff zones

Optional table.

Analogous to DIVA table FZ_TRANSITION_POINT:

*An intermediate point of a FARE_ZONE_TRANSITION. This indicates the fare zone(s) crossed (GER: Auxiliary points for tariff zone transitions)*

| | | Table: fare_zone_transition_point | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | fare_zone.version |
| | Y | Obligation (mandatory) | FARE_ZONE_TRANSITION_NR | Decimal (18) | | External key for tariff zone transition ID. Referenced by fare_zone_transition.din. | - |
| | Y | Obligation (mandatory) | CONSEC_NR | decimal (3) | 1..999 | Sequence of fare points along the route segment | - |
| | N | Optional | DISTANCE | decimal (6) | [m], in metres | Distance to previous fare point in the sequence | - |
| | N | Obligation (mandatory) | FARE_ZONE1_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set | fare_zone.fare_zone_nr |
| | N | Optional | FARE_ZONE2_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set | fare_zone.fare_zone_nr |
| | N | Optional | FARE_ZONE3_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set | fare_zone.fare_zone_nr |
| | N | Optional | FARE_ZONE4_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set | fare_zone.fare_zone_nr |
| | N | Optional | FARE_ZONE5_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set | fare_zone.fare_zone_nr |
| | N | Optional | FARE_ZONE6_NR | decimal (5) | | Fare zone, tariff zone, -1 or blank means not set | fare_zone.fare_zone_nr |

*Table 21 Fare_Zone_Transition_Point - Auxiliary points for fare zone transitions*

**Ex:**

```
VERSION;FARE_ZONE_TRANSITION_NR;CONSEC_NR;DISTANCE;FARE_ZONE1_NR;FARE_ZONE
2_NR;FARE_ZONE3_NR;FARE_ZONE4_NR;FARE_ZONE5_NR;FARE_ZONE6_NR;
 1;    0001;001;       0;7050;7051;  -1;  -1;  -1;  -1;
 1;    0001;002;       0;5040;7051;  -1;  -1;  -1;  -1;
 1;    0002;001;       0;6074;6309;  -1;  -1;  -1;  -1;
 1;    0002;002;       0;6076;6309;  -1;  -1;  -1;  -1;
 1;    0003;001;       0;6074;6309;  -1;  -1;  -1;  -1;
 1;    0003;002;       0;6076;6309;  -1;  -1;  -1;  -1;
...
```

## 5.5 Line / network / operating data
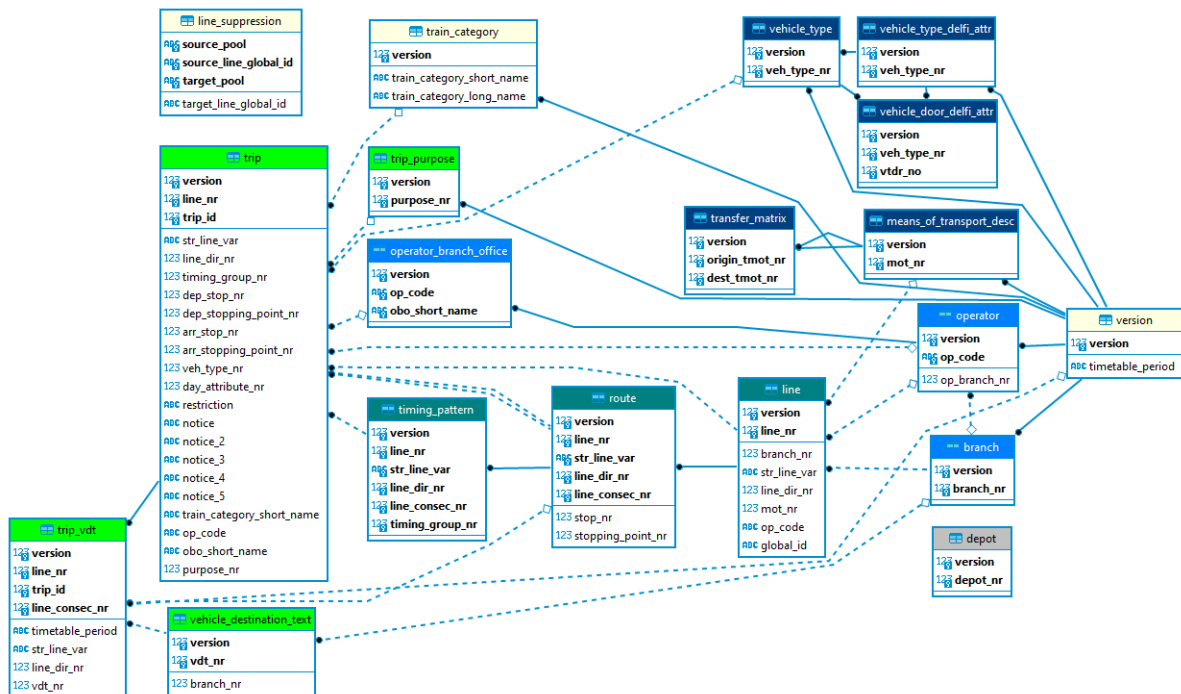


*Figure 7 ER diagram line, network, operational data*

### 5.5.1 means_of_transport_desc.din

Means of transport and connecting means of transport

Corresponds to the DIVA transport texts. They are used to label means of transport in the EFA. Means of transport texts are also linked to the connecting means of transport / GIS means of transport that are predefined in DIVA and thus have an effect on the georeferencing of the timetable data in DIVA.

| | | Table: means_of_transport_desc | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | - | version.version |

| | | Table: means_of_transport_desc | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | MOT_NR | decimal (2) | | Number of the means of transport | - |
| | N | Obligation (mandatory) | MOT_NAME | char (20) | | Name of the means of transport | - |
| | N | Obligation (mandatory) | TMOT_NR | decimal (2) | | Number of the connecting means of transport<br><br>0 .. Train<br><br>1 .. Commuter railway<br><br>2 .. Underground railway<br><br>3 .. City railway<br><br>4 .. Tram<br><br>5 .. City bus<br><br>6.. Regional bus<br><br>7 .. Express bus<br><br>8 . Cableway, rack railway<br><br>9 . Ship<br><br>10 .. AST on-demand bus<br><br>11 .. Other<br><br>12 .. Airplane<br><br>13 .. Train (local traffic)<br><br>14 .. Train (long distance)<br><br>15 .. Train (long distance with surcharge)<br><br>16 .. Train (long distance with special fare)<br><br>17 .. Rail replacement service ()<br><br>18 .. Train shuttle<br><br>19 .. Citizens' bus | - |
| | N | Optional | TMOT_NAME | Char(xx) | | Name of the connecting means of transport (hard-wired in DIVA)<br><br>See list above (TMOT_NR) | - |

*Table 22 Means_Of_Transport_Desc - Interchange mode of transport / GIS mode of transport*

Beschreibung_Austauschformat_DINO_V2.3.0_03.04.2023_published_en
21.03.23     50 Page of 104

### 5.5.2 transfer_matrix.din

Transfer matrix

The interchange matrix in DIVA defines interchange times between means of transport, unless they have been defined in DIVA footpath matrices (see table stop_footpath.din).

| Table: transfer_matrix | | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | - | means_of_transport_desc.version |
| | Y | Obligation (mandatory) | ORIGIN_TMOT_NR | decimal (2) | | Number of the means of transport of origin<br><br>0 .. Train<br><br>1 .. Commuter railway<br><br>2 .. Underground railway<br><br>3 .. City railway<br><br>4 .. Tram<br><br>5 .. City bus<br><br>6.. Regional bus<br><br>7 .. Express bus<br><br>8 . Cableway, rack railway<br><br>9 . Ship<br><br>10 .. AST on-demand bus<br><br>11 .. Other<br><br>12 .. Airplane | means_of_transport.orig_tmot_nr |
| | Y | Obligation (mandatory) | DEST_TMOT_NR | decimal (2) | | Number of the connecting means of transport<br><br>See also ORIGIN_TMOT_NR | means_of_transport.dest_tmot_nr |
| | N | Obligation (mandatory) | TIME | decimal (3) | | Time (in minutes)<br><br>May be empty, which means zero | - |

*Table 23 Transfer_Matrix - Transfer times between means of transport (fallback)*

The table must only contain the means of transport that are actually used in the subnetwork / data set. It serves as a generic fallback level if transfer times between areas have not been defined otherwise at a stop. Entries in Stop_Footpath.din always have priority.

### 5.5.3 vehicle_type.din

Vehicle types

Optional table.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| **Table: vehicle_type** | | | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | vehicle_type_delfi_attr.version, version.version |
| | Y | Obligation (mandatory) | VEH_TYPE_NR | decimal (8) | | Vehicle type number, AVL number is preferred if available | vehicle_type_delfi_attr.veh_type_nr |
| | N | Optional | VEH_TYPE_SEATS | decimal (3) | | Number of seats | - |
| | N | Optional | VEH_TYPE_STRAPS | decimal (3) | | Standing spaces | - |
| | N | Optional | PLACES_FOR_DISABLED_PERSONS | decimal (3) | | Disability-friendly spaces | - |
| | N | Optional | VEH_TYPE_TEXT | char (40) | | Type description | - |
| | N | Optional | STR_VEH_TYPE | char (4) | | Vehicle type abbreviation | - |
| | N | Optional | VEH_TYPE_DOOR_WIDTH | decimal(4) | | Vehicle door width in mm | - |
| | N | Optional | VEH_TYPE_WIDTH | decimal(4) | | Vehicle width in mm | - |
| | N | Optional | VEH_TYPE_HEIGHT | decimal(4) | | Vehicle height in mm | - |
| | N | Optional | VEH_TYPE_ACCESS_EQUIP | decimal(1) | | Boarding aids included<br><br>0 = No lift equipped vehicle<br><br>BIT1 = 1 = Has lift<br><br>BIT2 = 2 = Has ramp<br><br>BIT3 = 4 = Has multi purpose space | - |

| Table: vehicle_type | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | N | Optional | VEH_DELFI_TYPE | char(10) | | New from DINO Version 2.3<br><br>Reference to a DELFI FC type (especially bus types)<br><br>These are described in a separate DELFI document | - |

*Table 24 Vehicle_Type - Optional information on vehicle types*

**Ex:**

```
VERSION;VEH_TYPE_NR;VEH_TYPE_SEATS;VEH_TYPE_STRAPS;HANDICAP_PLACES;VEH_TYPE_TEXT;STR_VEH_TYPE
1;2;85;143"0;"Cityf"e"";"""F8"
1;5;101;151"0;"Comb"n"";"""F8"
1;9;60;96"0;"""T"";"""T6"
```

### 5.5.4    vehicle_type_delfi_attr.din

**New from DINO 2.3**

DELFI - Attributes of the vehicle types

This table is optional. It contains the DELFI accessibility characteristics of a vehicle type

| Table: vehicle_type_delfi_attr | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | int (2) | | Basic version | Vehicle_type.version |
| | Y | Obligation (mandatory) | VEH_TYPE_NR | int (8) | | Vehicle type number | Vehicle_type.VEH_TYPE_NR |
| | | | D_3010 | boolean | 0..1 | Vehicle floor plan available | - |
| | | | D_3020 | boolean | 0..1 | Display of the next stop/station and destination in the vehicle interior | - |
| | | | D_3021 | boolean | 0..1 | Display of line number and destination on the front, sides and/or rear of the vehicle | - |
| | | | D_3030 | boolean | 0..1 | Automatic announcements (inside) | - |
| | | | D_3031 | boolean | 0..1 | Automatic announcements (outside) | - |
| | | | D_3040 | char(100) | | Type of door opening: #w | - |
| | | | D_3041 | Int (3) | 0..999 | Clear width vehicle door: #w cm | - |
| | | | D_3050 | boolean | 0..1 | Seats available | - |

| | | Table: vehicle_type_delfi_attr | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | | | D_3051 | char(100) | | Location of the seats: #w | - |
| | | | D_3060 | boolean | 0..1 | Toilet available | - |
| | | | D_3061 | boolean | 0..1 | Wheelchair accessible toilet available | - |
| | | | D_3070 | boolean | 0..1 | Ticket machine available | - |
| | | | D_3080 | Int(3) | 0..999 | Entry gap: #w cm<br><br>Is calculated in DIVA / EFA per stop, therefore not available here. | - |
| | | | D_3090 | Int(3) | 0..999 | Vehicle width in the door area: #w cm | - |
| | | | D_3100 | Int(3) | 0..999 | Height of the vehicle floor: #w cm | - |
| | | | D_3120 | boolean | 0..1 | Ramp available | - |
| | | | D_3121 | Int(3) | 0..999 | Ramp length: #w cm | - |
| | | | D_3122 | Int(3) | 0..999 | Width of the ramp: #w cm | - |
| | | | D_3123 | Int(3) | 0..999 | Load capacity of the ramp: #w kg | - |
| | | | D_3124 | boolean | 0..1 | Ramp inside the vehicle | - |
| | | | D_3125 | Int(3) | 0..999 | Width at the climb when using the ramp: #w cm | - |
| | | | D_3126 | Int(3) | 0..999 | Length on the climb when using the ramp: #w cm | - |
| | | | D_3127 | Int(3) | 0..100 | Ramp inclination: #w %<br><br>Attention export only - calculated from length and entry height | - |
| | | | D_3130 | boolean | 0..1 | Lift available | - |
| | | | D_3131 | Int(3) | 0..999 | Width at the riser when using the lift: #w cm | - |
| | | | D_3132 | Int(3) | 0..999 | Length on the climb when using the lift: #w cm | - |
| | | | D_3133 | Int(3) | 0..999 | Lifting capacity lift: #w kg | - |
| | | | D_3140 | boolean | 0..1 | Multipurpose area available | - |

*Table 25 Vehicle_Type_Delfi_Attr - Optional accessibility properties of a vehicle type*

### 5.5.5 vehicle_door_delfi_attr.din

**NEW from DINO 2.3**

DELFI - Attributes of the doors per vehicle type

Some DELFI accessibility properties of a vehicle type can (and must) be set in DIVA per door of a vehicle type.

**Table: vehicle_door_delfi_attr**

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | Y | Obligation (mandatory) | VERSION | int (2) | | Basic version | vehicle_type.version |
| | Y | Obligation (mandatory) | VEH_TYPE_NR | int (8) | | Vehicle type number | vehicle_type.veh_type_nr |
| | Y | Obligation (mandatory) | VTDR_NO | Int(2) | 1..99 | No. Door in vehicle | - |
| | | Obligation (mandatory) | VTDR_POSITION | int | 0..99999 | Distance (cm) of the door centre from the front of the vehicle | - |
| | | | D_3041 | Int (3) | 0..999 | Clear width vehicle door: #w cm (overrides value from vehicle type) | - |
| | | | D_3101 | Int(3) | 0..999 | Height of the lowest entry step: #w cm | - |
| | | | D_3110 | boolean | 0..1 | Stage present (Export only, D:3112 Number of stages > 0) | - |
| | | | D_3111 | Int(3) | 0..999 | Step height: #w cm | - |
| | | | D_3112 | int | 1 | Number of steps in the entrance area: #w | - |
| | | | D_3113 | boolean | 0..1 | Foldable / variable access steps | - |
| | | | VTDR_BICYCLE | boolean | 0..1 | Use with bicycle possible | - |
| | | | VTDR_PRAM | boolean | 0..1 | Use with pram possible | - |
| | | | VTDR_WHEELCHAIR | boolean | 0..1 | Use with wheelchair possible | - |

*Table 26 Vehicle_Door_Delfi_Attr - DELFI attributes of doors per vehicle type*

### 5.5.6 operator.din

Operator

Optional table.

**Table: operator**

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version VDV 452::Basic version | version.version, branch.version, operator_branch_office.version |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| **Table: operator** | | | | | | | |
|  | Y | Obligation (mandatory) | OP_CODE | char(10) | > 0 | Operator N VDV 452::UNTERNEHMEN r | check operator_branch_office.op_code |
|  | N | Optional | OP_BRANCH_NR | decimal (6) |  | Default operating branch | branch.op_branch_nr |
|  | N | Optional | OP_SHORT_NAME | char(7) |  | Short designation, VDV 452::ABK_UNTERNEHMEN | - |
|  | N | Obligation (mandatory) | OP_LONG_NAME | char(255) |  | Long designation | - |
|  | N | Optional | OP_PUBLIC_SHORT_NAME | char (7) |  | Representation in passenger information | - |
|  | N | Optional | OP_LICENCE_NAME | char (255) |  | Licence name | - |
|  | N | Optional | OP_TRADING_NAME | char (255) |  | Trading name, used e.g. in marketing | - |
|  | N | Optional | OP_VAT_REGISTERED_FLAG | decimal(1) | 0,1 | VAT / turnover tax Registration | - |

*Table 27 Operator - Optional information on entrepreneurs, enforces Operator_Branch_Office if applicable*

An operator must have at least one branch office. The operator.din table must therefore always be delivered together with operator_branch_office.din.

DIVA data suppliers are defined via the DIVA import configuration and have therefore not been included in the operator.din table.

**Ex:**

```
VERSION;OP_CODE;OP_BRANCH_NR;OP_SHORT_NAME;OP_LONG_NAME;OP_PUBLIC_SHORT_NA
ME;OP_LICENCE_NAME;OP_TRADING_NAME;OP_VAT_REGISTERED_FLAG;
 1;01;;PT;ÖBB Postbus GmbH;Pt;;;0;
 1;25;;LIEm;LIECHTENSTEINmobil;;;;0;
 1;27;;BUS OAG;BUS Ostschweiz AG;;;;0;
```
…

### 5.5.7    operator_branch_office.din

Operator branches

Conditionally optional table. However, if the operator.din table is supplied, operator_branch_office.din must also be included in the data set.

| | Table: operator_branch_office | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | operator.version |
| | Y | Obligation (mandatory) | OP_CODE | char(10) | | Foreign key for operator.din::OP_Code | operator.op_code |
| | Y | Obligation (mandatory) | OBO_SHORT_NAME | char (10) | | Short name of the branch, unique within one operator | - |
| | N | Optional | OBO_INTERNAL_PHONE | char (50) | | Internal telephone number | - |
| | N | Optional | OBO_PUBLIC_PHONE | char (50) | | Public phone number | - |
| | N | Optional | OBO_FAX_NR | char (50) | | Fax number | - |
| | N | Optional | OBO_ADDRESS | char (500) | | Address (street address of the operator) | - |
| | N | Optional | OBO_CONTAC_ADDRESS | char (500) | | Public address (e.g. helpdesk) | - |
| | N | Optional | OBO_URL | char (255) | | Website | - |

*Table 28 Operator_Branch_Office - Contact information of a contractor*

Depending on the data situation in DIVA, the OBO_SHORT_NAME can also be filled by OBO_EXT_Code for the export.

**Ex:**

```
VERSION;OP_CODE;OBO_SHORT_NAME;OBO_INTERNAL_PHONE;OBO_PUBLIC_PHONE;OBO_FAX
_NR;OBO_ADDRESS;OBO_CONTAC_ADDRESS;OBO_URL;
 1;01;Al;01/71101;01/71101;;Servicetelefon Wien;;
 1;01;BZ;05552/62746;05552/62746;;ÖBB   Postbus   GmbH,   Äuleweg   126700
Bludenz;;
 1;01;LA;05442/64422;T 05442/64422;;ÖBB Postbus GmbH, Buntweg 46511 Zams;;
 1;25;VA;+423 237 94 94;T +423 237 94;;LIECHTENSTEINmobil, Städtle 17FL-
9490 Vaduz;;
```

### 5.5.8   depot.din

Depots

Optional table. Currently not evaluated by the import.

| | Table: depot | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | - | - |
| | Y | Obligation (mandatory) | DEPOT_NR | decimal (5) | | - | - |
| | N | Optional | DEPOT_TEXT | char (40) | | - | - |
| | N | Optional | DEPOT_ABBREV | char (5) | | - | - |

*Table 29 Depot - optional table for depots, currently not evaluated*

**Ex:**

```
VERSION;DEPOT_NR;DEPOT_TEXT;DEPOT_ABBREV
1;1001; "Hofwiesengasse"; "H1"
1;1002; "Wolfganggasse"; "W1"
```

### 5.5.9   branch.din

Operating branch information

| | Table: branch | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | version.version |
| | Y | Obligation (mandatory) | BRANCH_NR | decimal (2) | 0..99 | Company branch number, RBL number is preferred if available | - |
| | | Optional | STR_BRANCH_NAME | char (6) | | Short designation of the operating branch | - |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | | **Table: branch** | | | | | |
| | | Obligation (mandatory) | BRANCH_NAME | Char (40) | | Name of the operating branch | - |

*Table 30 Branch - Operating branches*

**Ex:**

```
VERSION;BRANCH_NR;STR_BRANCH_NAME;BRANCH_NAME;
1;10;Linz;City of Linz;
1;16;RV_PB;RV Postbus;
1;13;RV_Pri;RV Private;
```

### 5.5.10   timing_pattern.din

Service journey pattern-specific travel and stopping times

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | Foreign key |
|---|-----|-----------------|----------------|-----------|-------------|----------|-------------|
| | | **Table: timing_pattern** | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | route.version |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | route.line_nr |
| | Y | Obligation (mandatory) | STR_LINE_VAR | char (4) | | Route number | route.str_line_var |
| | Y | Obligation (mandatory) | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | route.line_dir_nr |
| | Y | Obligation (mandatory) | LINE_CONSEC_NR | decimal (3) | | Serial stopping point number in the service journey pattern | route.line_consec_nr |
| | Y | Obligation (mandatory) | TIMING_GROUP_NR | decimal (3) | 1 .. 999 | Travel time group number | - |
| | N | Obligation (mandatory) | TT_REL | decimal (6) | | Travel time in seconds from the previous stop, -1 if passing through i.e. 1st entry on the route is 0 because there is no preceding stop. | - |
| | N | Obligation (mandatory) | STOPPING_TIME | decimal (6) | | Stopping time in seconds | - |

*Table 31 Timing_Pattern - Route-dependent travel and stop times*

### 5.5.11 route.din

Points in sequence

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| **Table: route** | | | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | **line.version** |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | **line.line_nr** |
| | Y | Obligation (mandatory) | STR_LINE_VAR | char (4) | | Route number | **line_str_line_var** |
| | Y | Obligation (mandatory) | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | **line.line_dir_nr** |
| | Y | Obligation (mandatory) | LINE_CONSEC_NR | decimal (3) | | Serial stopping point number in the service journey pattern | - |
| | N | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | Internal stop number | - |
| | N | Obligation (mandatory) | STOPPING_POINT_NR | decimal (2) | | Stopping point number | - |

| | | | Table: route | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Obligation (mandatory) | STOPPING_POINT_TYPE | decimal (2) | | Stop type: -1 = passage = \| 0 = Normal stop 1 = Demand stop = B 2 = boarding ban = A 3 = Disembarkation ban = E 4 = no service in the city = 5 = unproduct "v "no passenger "m 6 = with bicycle only get off = 7 = with bicycle only boarding = FE 8 = with bicycle no service in town = F 9 = Operating stop for course wagons (change of suspension, NoBoardingAndAlighting) = K 10 = Operating port "Diensth "lt" (NoBoardingAndAlighting) = T 11= Demand stop, only exit = BA 12 = Demand stop, only entry = BE Types 2 to 4 are not filled if the service_interdiction.din (22) table is filled The attribute value "5 = No passenger transport" can be used to filter for unproductive trip segments. | - |
| | N | Optional | LENGTH | decimal (7) | | Distance to previous stop in metres, -1 = not filled. i.e. the first entry in LID_COURSE for each route is 0. | - |

| | Table: route | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Optional | PRINT_FLAG | decimal (7), NULL | | NULL = empty if not route 0<br><br>BIT1 = Do not print stop<br><br>BIT2 = Horizontal line<br><br>BIT3 = Raster<br><br>BIT4 = Blank line<br><br>BIT5 = Omission<br><br>BIT6 = combine on/off times<br><br>BIT7 = thick<br><br>BIT8 = in the special issue<br><br>BIT9 = not in the special issue<br><br>BIT10 = not in the special issue<br><br>BIT11 = dashed line<br><br>BIT12 = Highlight stop (bold font)<br><br>BIT13 = Highlight stop (italic font)<br><br>BIT14 = Integrate assigned stops | - |
| | N | Optional | PRINT_FLAG_SSTT | decimal (7), NULL | | NULL = empty if not route 0<br><br>BIT1 = Intermediate target<br><br>BIT2 = Notional final target<br><br>BIT3 = Breaking point<br><br>BIT4 = Ring line start<br><br>BIT5 = End of ring line<br><br>BIT6 = Ring Line Separation Point<br><br>BIT7 = bead line branch point<br><br>BIT8 = Stop name with location<br><br>BIT9 = Stop name without location<br><br>BIT10 = place name only | - |

| | | Table: route | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | N | Optional | OPTION_FLAG | decimal (7), NULL | | NULL = empty if not route 0<br><br>BIT1 = Not foldable after this stop<br><br>BIT2 = Change direction of travel (make head)<br><br>BIT3 = Print stop in course timetable<br><br>BIT4 = Mini Timetable: Important stop<br><br>BIT5 = Posting timetable: Highlight stop<br><br>BIT6 = bead string: Stop name with place<br><br>BIT7 = bead string: Stop name without locality<br><br>BIT8 = bead string: Place name only<br><br>BIT9 = AHF Print code: Do not print stop<br><br>BIT10 = AHF Print code: Print stop only if there is space<br><br>BIT11 = Print indicator for ZOB: Important stop for ZOB<br><br>BIT12 = Duty Intermediate Destination Flag | - |

*Table 32 Route - Travel paths*

The following applies to the DIVA import: if the *STOPPING_POINT_NR* is 0, and if there is no entry in rec_stopping_point, then the stop is mapped to itself.

The specification of a "**zero route**" was defined for various third-party applications, e.g. for counting passengers. The definition and use of the zero travel distance are specified here:

The DIVA margin column can be output as a route with the number 0. This service journey pattern is not actually travelled and all the trips can be mapped onto it in chronologically ascending order.

When exchanging DINO timetable data, care must be taken to ensure that the trip reference always refers to the actual routes. This also applies in particular to route-specific elements such as destination texts, announcement texts, etc. Also, LINE_CONSEC_NR always refers to the actual service journey pattern and NOT to the zero route.

### 5.5.12   trip_purpose.din

Trip purposes, trip types

Not evaluated by the import

| | Table: trip_purpose | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Version | version.version |
| | Y | Obligation (mandatory) | PURPOSE_NR | decimal (20) | | purpose number | - |
| | N | Obligation (mandatory) | PURPOSE_TEXT | char (40) | | Description | - |
| | N | Optional | STR_PURPOSE | char (5) | | Abbreviation | - |

*Table 33 Trip_Purpose - trip types, not evaluated by DIVA import*

**Ex:**

```
VERSION;PURPOSE_NR; PURPOSE_TEXT; STR_PURPOSE
1;1; "IntermodalJourneyPlanner"; "BIT1"
1;2; "PersonalTimetable"; "BIT2"
1;3; "StopTimetable"; "BIT3"
```

The bits must be set as follows for the trips in trip.din:

```
trip.din:
Trip 1.PURPOSE_NR = 3
Trip 2.PURPOSE_NR = 6
Trip 3.PURPOSE_NR = 4
```

A bit-by-bit evaluation then results in the following bits being set: Bin: 110 = decimal: 6 = BIT2|BIT3

### 5.5.13 line.din

Line directory

| | Table: line | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | branch.version, means_of_transport_desc.version, operator.version |
| | N | Obligation (mandatory) | BRANCH_NR | decimal (2) | 0..99 | Designation of the operating branch or authority | branch.branch_nr |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | - |
| | Y | Optional | STR_LINE_VAR | char (4) | | Route number | - |
| | N | Optional | LINE_NAME | char (40) | | Published line number | - |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | Y | Optional | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | - |
| | N | Optional | LAST_MODIFIED | char (20) | | Last modification date of the line including time stamp | - |
| | N | Optional | MOT_NR | decimal (2) | | Number of the means of transport | means_of_transport_desc.mot_nr |
| | N | Optional | VALID_FROM | decimal(8) | YYYYMMDD | Validity of the line<br><br>Currently only supported by DINO export | - |
| | N | Optional | VALID_TO | decimal(8) | YYYYMMDD | Validity of the line<br><br>Currently only supported by DINO export | - |
| | N | Optional | OP_Code | Char(10) | | Foreign key for operator.din::OP_Code | operator.op_code |
| | N | Optional | OBO_SHORT_NAME | char(10) | | Foreign key for operator_branch_offic.din:: OBO_SHORT_NAME | - |
| | N | Optional | ROUTE_TYPE | decimal (2) | | Trip purpose type, trip type | - |
| | N | Optional | GLOBAL_ID | Char (128) | ISO 8859-1 | | - |

| | | Table: line | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Optional | BIKE_RULE | decimal(2) | | Contains the bike-loading rule. Bicycle transport rules: -1 = NoBicycle, 0 = VVS_Rail, 1 = VVS_CityRail, 2 = MVV, 3 = DB, 4 = GM, 5 = IV, 6 = TF, 7 = VVS_END, 8 = AlwaysAllowed, 9 = RegulatedPerJourney, 10 = CTA_Rail_Chicago, 11 = SundaysAndBankHolidays, 12 = DART, 13 = GVH_RegioBus, 14 = AT_SVV, 15 = AT_VOR_Subway, 16 = VVS_BusRegionalFrom_09, 17 = VVS_BusRegionalWithBlockingTimes | - |

| Table: line | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandator y field | Attribute name | Data type | Value range | Comments | FK |
| | N | Optional | LINE_SPECIAL _FARE | decimal(1) | | Tariff specifics:<br><br>0 = Line outside the interconnected area,<br><br>1 = Line in the interconnected area,<br><br>2 = Line in the network area but no fare calculation,<br><br>3 = Compound line subject to surcharge,<br><br>4 = Compulsory supplementary line but no fare calculation<br><br>5 = ICE line | - |
| | N | Optional | LINE_SHORT_ NAME | Char (3) | ISO 8859-1 | New in DINO22. Unique DIVA4 line name (lower case only) | - |
| | N | Optional | LINE_SUFFIX | Char (1), empty | ISO 8859-1 | New in DINO22. Unique DIVA4 line completion (lower case only) | - |
| | N | Optional | LV_VERSION | decimal(1) | | New in DINO22. Unique DIVA4 line version number | - |

*Table 34 Line - List of lines*

The *LINE_NR* defined in DINO must be unique across all operating branches.

Means of transport are only supported by the DINO export. All routes of a line must have the same means of transport and the same *LINE_NAME*.

The attributes VALID_FROM and VALID_TO refer to the validity of the DIVA 4 line version. However, the DINO data can also be exported on the DIVA side in such a way that the validities are converted into the traffic restrictions of the individual journeys. If you have any questions, please contact the MENTZ GmbH Support unit.

**Ex:**

```
VERSION;BRANCH_NR;BRANCH_NAME;LINE_NR;STR_LINE_VAR;LINE_NAME;LINE_DIR_NR;LAST_MODIFIED;
1;16;RV Postbus;1120016;101;112;1;;
1;16;RV Postbus;1120016;102;112;1;;
1;16;RV Postbus;1120016;201;112;2;;
1;16;RV Postbus;1120016;202;112;2;;
```

**GLOBAL_ID**

The GLOBAL_ID is a unique ID for identifying line objects. If this optional field is filled, the provider must ensure that it is unique.

**LINE_SPECIAL_FARE**

Serves to map special tariff features and has an effect on the fare calculation of the supplied tariff calculation programme, e.g. EFA.

### 5.5.14 vehicle_destination_text.din

This table is optional.

**DINO description ([overview of all relations](#))**

List of destinations displayed on/in the vehicle

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | branch.version |
| | Y | Optional | BRANCH_NR | decimal (2) | 0..99 | Designation of the operating branch or authority | branch.branch_nr |

| | | Table: vehicle_destination_text | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VDT_NR | decimal (8) | 0..99999999 | Vehicle destination text number Attention: The VDT_NR can hold numeric values with up to eight digits. However, the length of the values to be transferred must be adapted to suit the value ranges of the respective target system. VDT_NR = 0 clears the display | - |
| | N | Optional | VDT_TEXT_DRIVER1 | char (160) | ISO 8859-1 | Driver destination text display line1 | - |
| | N | Optional | VDT_TEXT_DRIVER2 | char (160) | ISO 8859-1 | Driver destination text display line2 | - |
| | N | Optional | VDT_TEXT_FRONT1 | char (160) | ISO 8859-1 | Front display text line 1 | - |
| | N | Optional | VDT_TEXT_FRONT2 | char (160) | ISO 8859-1 | Front display text line 2 | - |
| | N | Optional | VDT_TEXT_FRONT3 | char (160) | ISO 8859-1 | Front display text line 3 | - |
| | N | Optional | VDT_TEXT_FRONT4 | char (160) | ISO 8859-1 | Front display text line 4 | - |
| | N | Optional | VDT_TEXT_SIDE1 | char (160) | ISO 8859-1 | Lateral destination display text line 1 | - |
| | N | Optional | VDT_TEXT_SIDE2 | char (160) | ISO 8859-1 | Lateral destination display text line 2 | - |
| | N | Optional | VDT_TEXT_SIDE3 | char (160) | ISO 8859-1 | Lateral destination display text line 3 | - |
| | N | Optional | VDT_TEXT_SIDE4 | char (160) | ISO 8859-1 | Lateral destination display text line 4 | - |
| | N | Optional | VDT_LONG_NAME | char (160) | ISO 8859-1 | Name of the vehicle destination text | - |
| | N | Optional | VDT_SHORT_NAME | char (68) | ISO 8859-1 | Short name of the vehicle destination text | - |
| | N | Optional | VDT_TEXT_INTERIOR1 | char (160) | ISO 8859-1 | Text of the inside display line 1 Name correction of VDT_TEXT_INNEN | - |
| | N | Optional | VDT_TEXT_INTERIOR2 | char (160) | ISO 8859-1 | Text of the interior display line 2 | - |
| | N | Optional | VDT_TEXT_INTERIOR3 | char (160) | ISO 8859-1 | Text of the interior display line 3 | - |
| | N | Optional | VDT_TEXT_INTERIOR4 | char (160) | ISO 8859-1 | Text of the interior display line 4 | - |

*Table 35 Vehicle_Destination_Text - Vehicle destination texts in/on the vehicle*

### 5.5.15 trip_vdt.din

**DINO description ([overview of all relations](#))**

Stop or route-segment-related vehicle destination texts

The trip_vdt table is only an allocation table, similar to JOURNEY_VDT in DIVA4. It contains references to the line and stops, as well as the table vehicle_destination_text

| | | | Table: trip_vdt | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | route.version, trip.version, vehicle_destination_text.version version.version |
| | Y | Optional | TIMETABLE_PERIOD | char (4) | | Timetable project | version.timetable_period |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | route.line_nr, trip.line_nr |
| | N | Optional | STR_LINE_VAR | char (4) | | Route number | route.str_line_var |
| | N | Optional | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | route.line_dir_nr |
| | Y | Obligation (mandatory) | TRIP_ID | decimal (8) | | Internal trip number - only if trip-specific destination text | trip.trip_id |
| | Y | Obligation (mandatory) | LINE_CONSEC_NR | decimal (3) | | Consecutive stopping point number in the service journey pattern | route.line_consec_nr |
| | N | Optional | STOP_NR | decimal (5) | 0.99999 | &lt;correction&gt; in V 2.3 → was already part of the DINO exports | FK to stop_point |
| | N | Optional | STOPPING_POINT_NR | decimal (2) | 0..99 | &lt;correction&gt; in V 2.3 → was already part of the DINO exports | FK to stop_point &lt;2DO&gt; |

| | Table: trip_vdt | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Obligation (mandatory) | VDT_NR | decimal (8) | 0..99999999 | Vehicle destination text number Attention: The VDT_NR can hold numeric values with up to eight digits. However, the length of the values to be transferred must be adapted to suit the value ranges of the respective target system. VDT_NR = 0 clears the display | vehicle_destination_text.vdt_nr |

*Table 36 Trip_VDT - Stop/route-related vehicle destination texts*

For stop-related notices, only those stopping points must be specified (LINE_CONSEC_NR) at which the destination text changes.

### 5.5.16   train_category.din

**DINO description ([overview of all relations](#))**

Train categories

| | Table: train_category | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | version.version |
| | Y | Optional | TRAIN_CATEGORY_SHORT_NAME | char (255) | | Train category - short designation | - |
| | N | Obligation (mandatory) | TRAIN_CATEGORY_LONG_NAME | char (255) | | Train category - long designation / description | - |

*Table 37 Train_Category - train categories*

### 5.5.17   line_suppression.din

**DINO description ([overview of all relations](#))**

Line suppressions

Integration systems import line data from several sources. These sources may include multiple instances of the same lines. Integration systems can recognise these duplicates by examining the

global line ID. The line suppression table is used to provide the integration system with conflict resolution rules that are captured in the exporting system.

| | **Table: line_suppression** | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Mandatory | VERSION | Decimal(2) | 0..99 | FK to version | Version.version |
| | Y | Obligation (mandatory) | SOURCE_POOL | Char(30) | | Identifier of the source data pool | - |
| | Y | Obligation (mandatory) | SOURCE_LINE_GLOBAL_ID | Char (50) | ISO 8859-1 | Unique Identifier of the source line | - |
| | Y | Obligation (mandatory) | TARGET_POOL | Char(30) | | Identifier of the target data pool | - |
| | Y | Optional | TARGET_LINE_GLOBAL_ID | Char (50) | ISO 8859-1 | Unique Identifier of the target line | - |

*Table 38 Line_Suppressi-n - Line suppressions*

The columns SOURCE_POOL and TARGET_POOL refer to DINO data deliveries.

An exporting DIVA system enters the network here and an exporting IVU.pool system can probably specify the provider.

If DINO data is being imported, the importing system must note the corresponding pool. An importing DIVA system imports a provider's data into a network, i.e. the network identifier corresponds to the pool in DIVA. In IVU.pool systems one network is probably assigned to one provider.

During the integration process, the DIVA system checks for each line whether there is a data record that contains this line as a SOURCE line key and whether there is also a line in the TARGET_POOL (network) with the TARGET_LINE_GLOBAL_ID.

By omitting the TARGET_LINE_GLOBAL_ID, you can ensure that the integration process only checks if there is data from the TARGET_POOL. If this is the case, the SOURCE_LINE is suppressed.
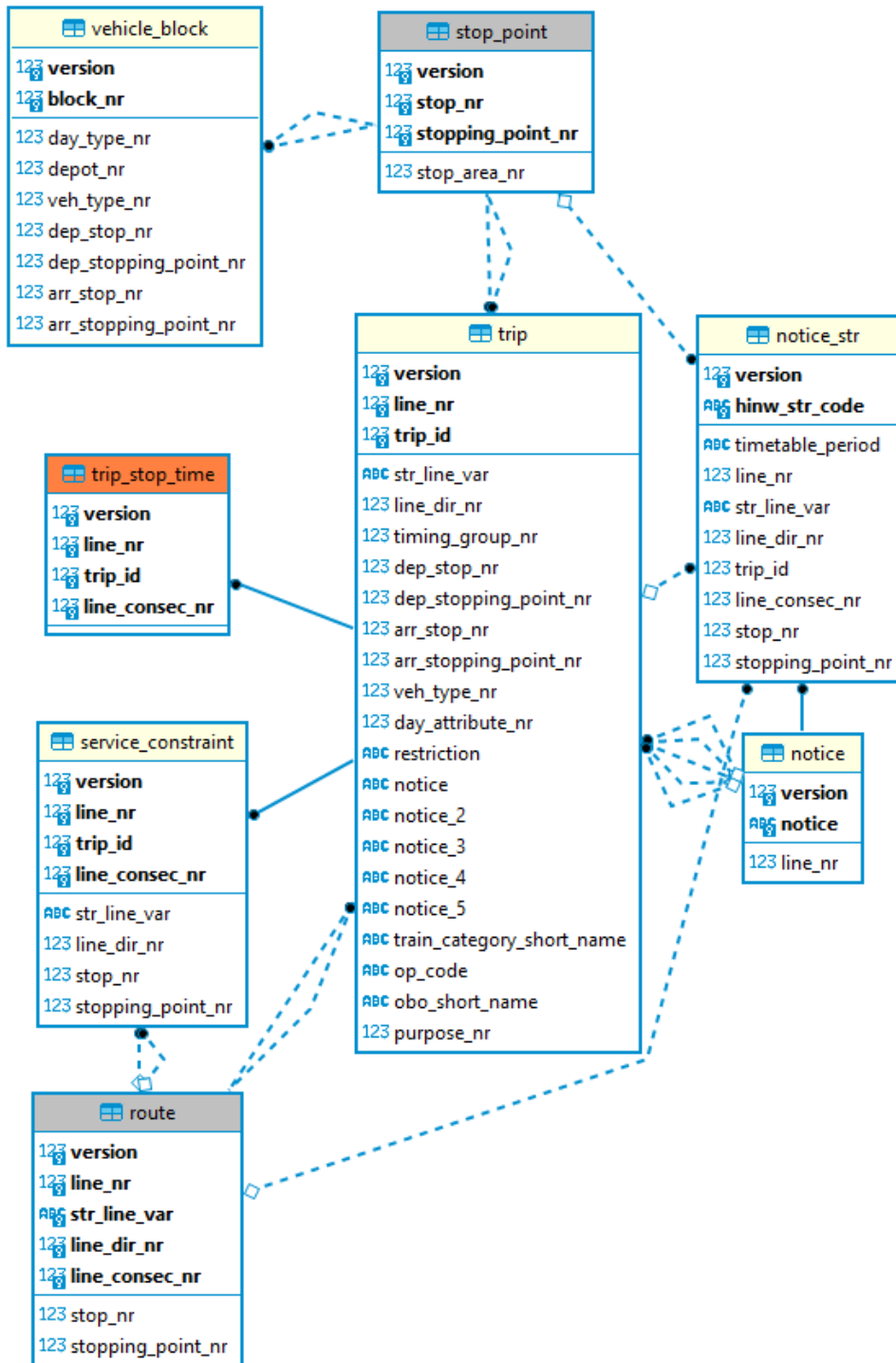
## 5.6 Timetable data



*Figure 8 ER diagram timetable data*

### 5.6.1 trip.din

VDV452 requires that the TRIP_ID alone is unique. The DINO standard is softer here: LINE_NR and TRIP_ID must be unique together.

The *DAY_ATTRIBUTE_NR* provided must also be defined in the day_attribute.din table.

**DINO description ([overview of all relations](#))**

Journeys

| Table: trip | | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | Together with all other FKs |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | Line, restriction, route |
| | N | Obligation (mandatory) | STR_LINE_VAR | char (4) | | Route number | Line, route |
| | N | Obligation (mandatory) | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | line, route |
| | N | Obligation (mandatory) | TIMING_GROUP_NR | decimal (3) | | Timing group number | Timing_pattern |
| | Y | Obligation (mandatory) | TRIP_ID | decimal (8) | | Internal trip number | n/a |
| | N | Optional | TRIP_ID_PRINTING | decimal (7) | | Trip number for printing | - |
| | N | Obligation (mandatory) | DEPARTURE_TIME | decimal (6) | | Departure time in seconds | - |
| | N | Obligation (mandatory) | DEP_STOP_NR | decimal (5) | 1..99999 | Internal departure stop number | Stop_point, route |
| | N | Obligation (mandatory) | DEP_STOPPING_POINT_NR | decimal (2) | | Departure stopping point number | Stop_point, route |
| | N | Obligation (mandatory) | ARR_STOP_NR | decimal (5) | 1..99999 | Internal arrival stop number | Stop_point, route |
| | N | Obligation (mandatory) | ARR_STOPPING_POINT_NR | decimal (2) | | Arrival stopping point number | Stop_point, route |
| | N | Optional | VEH_TYPE_NR | decimal (8) | | Vehicle type | Vehicle_type |
| | N | Obligation (mandatory) | DAY_ATTRIBUTE_NR | decimal (5) | | Day-type attribute number | Day_attributes |
| | N | Optional | RESTRICTION | char (10) | | Service restriction, also empty  Traffic restriction in DIVA up to 10-points.  Change in version 2.3: Extension from char(5) to char(10) | service_restriction.restriction |
| | N | Optional | NOTICE | char (5) | | Key to the notice table | notice |
| | N | Optional | NOTICE_2 | char (5) | | Key to the notice table | Notice |
| | N | Optional | NOTICE_3 | char (5) | | Key to the notice table | Notice |
| | N | Optional | NOTICE_4 | char (5) | | Key to the notice table | Notice |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| **Table: trip** | | | | | | | |
| | N | Optional | NOTICE_5 | char (5) | | Key to the notice table | notice |
| | N | Optional | ROUND_TRIP_ID | decimal (8) | | FK for circulation number (Block_No) | Vehicle_block.block_nr |
| | N | Optional | TRAIN_NR | decimal (8) | | Train or bus number | - |
| | N | Optional | TRAIN_CATEGORY_SHORT_NAME | char (10) | | Train category<br><br>Selected values:<br><br>BUS<br><br>EN<br><br>IC<br><br>OEC<br><br>OIC<br><br>R<br><br>REX<br><br>RJ<br><br>ZUG<br><br>EZ<br><br>Please refer to the list of train categories in the appendix of this document | Train_category (together with version) |
| | N | Optional | TRIP_EXT_KEY | char (50) | | External trip number (not used in DIVA)<br><br>This parameter was introduced for a third party but is not supported in DIVA through import or export. | - |
| | N | Optional | OP_CODE | char(10) | | Foreign key for operator.din::OP_Code | Operator_branch_office (& operator) |
| | N | Optional | OBO_SHORT_NAME | char(10) | | Foreign key for operator_branch_offic.din:: OBO_SHORT_NAME | Operator_branch_office |
| | N | Optional | GLOBAL_ID | Char (128) | ISO 8859-1 | A combination of two keys. GLOBAL_ID of the line + REAL_TIME_ID of the trip (real-time ID). From version 2.2 field length extended from Char(100) to Char(128). | - |
| | N | Optional | BIKE_ALLOWED | Boolean | (0 \| 1) | | - |
| | N | Optional | PURPOSE_NR | decimal(20) | | Foreign key for trip_purpose.din::PURPOSE_NR | Trip_purpose |

*Table 39 Trip - Journeys*

**GLOBAL_ID**

The GLOBAL_ID is a unique ID for identifying trip objects. If this optional field is filled, the provider must ensure that it is unique. It is possibly only unique within the calendar day.

**DEP_STOP_NR, DEP_STOPPING_POINT_NR, ARR_STOP_NR, ARR_STOPPING_POINT_NR**

These stops must be included in route.din in the scheduled route of this journey.

If these stops are not the start and destination of the journey, then the route of this journey is only travelled in sections.

### 5.6.2 trip_stop_time.din

This table is optional.

**DINO description (overview of all relations)**

Waiting time of a specific trip:

The time a vehicle has to wait at a specific stopping point on a specific trip. This time overrides the stop_time and the lid_travel_stop_time.

| Table: trip_stop_time | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Version | trip.version |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | 1-99999999 | internal line number | trip.line_nr |
| | Y | Obligation (mandatory) | TRIP_ID | decimal (8) | 1-99999999 | Number of the trip | Trip.trip_id |
| | Y | Obligation (mandatory) | LINE_CONSEC_NR | decimal (3) | | Consecutive Number of Stop | ? - |
| | | Obligation (mandatory) | STOPPING_TIME | decimal (6) | | Stopping Time at the stop in seconds | - |

*Table 40 Trip_Stop_Time - Optional, trip-specific waiting times*

### 5.6.3 vehicle_block.din

Supported by DINO export since format version 2.2.
Not currently supported by DIVA 4 DINO import.

**DINO description (overview of all relations)**

Vehicle round trips

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | | | **Table: vehicle_block** | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Version | stop_point.version, depot.version, vehicle_type.version |
| | Y | Obligation (mandatory) | DAY_TYPE_NR | decimal (9) | | type of the day | |
| | Y | Obligation (mandatory) | DEPOT_NR | decimal (5) | | number of the depot | depot.depot_nr |
| | Y | Obligation (mandatory) | BLOCK_NR | decimal (8) | | round trip ID | |
| | N | Obligation (mandatory) | VEH_TYPE_NR | decimal (2) | | vehicle type | vehicle_type.veh_type_nr |
| | N | Obligation (mandatory) | DEP_STOP_NR | decimal (5) | 1..99999 | departure of the round trip | stop_pint.dep_stop_nr |
| | N | Obligation (mandatory) | DEP_STOPPING_POINT_NR | decimal (2) | | stopping point of the departure of the round trip | stop_point.dep_stopping_point_nr |
| | N | Obligation (mandatory) | BEGIN_OF_BLOCK | decimal (6) | | time of the beginning of the round trip in seconds | - |
| | N | Obligation (mandatory) | ARR_STOP_NR | decimal (5) | 1..99999 | arrival stop of the round trip | stop_point.arr_stop_nr |
| | N | Obligation (mandatory) | ARR_STOPPING_POINT_NR | decimal (2) | | arrival stopping point of the round trip | stop_point.arr_stopping_point_nr |
| | N | Obligation (mandatory) | END_OF_BLOCK | decimal (6) | | time of the end of the round trip in seconds | - |

Table 41 Vehicle_Block - Vehicle Circulations [not supported by DIVA import].

### 5.6.4    notice.din

**DINO description ([overview of all relations](#))**

Service information texts without reference to service days

| Table: notice | | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | line.version |
| | Y | Optional | LINE_NR | decimal (8) | | Internal line number, if empty the note applies to all lines | line.line_nr |
| | Y | Obligation (mandatory) | NOTICE | char(5) | | Notice abbreviation | - |
| | N | Obligation (mandatory) | NOTICE_TEXT | char (1000) | | Note text, \n is interpreted as pagination. In accordance with the CSV specification, newline characters are permitted provided they are enclosed in double inverted commas (″Text with Line break″).[3] Cf. chap. **Fehler! Verweisquelle konnte nicht gefunden werden. Fehler! Verweisquelle konnte nicht gefunden werden.** | - |
| | N | Optional | CONTENT_TYPE | decimal (3) | | 0 .. Other notice (default) 1.. Train name notice 2 .. On-demand bus notice 3 .. Bicycle loading notice 4 .. Track notice 5 .. Regional train 6 .. Driver text 7 .. Offer 8 .. Fare code | - |

---

[3] https://csv-spec.org/#terminology: "**Record** (or **Row**) - A collection of fields. This is often referred to as a "line", but a single record can span multiple text lines if a field within it contains one or more line breaks. ... Line **Break** - Line breaks in CSV files can be CRLF (\r\n), LF (\n), and even in rare cases CR (\r)."

| | Table: notice | | | | | | |
|---|---|---|---|---|---|---|---|
| N | Optional | DISPLAY_TYPE | decimal (3) | | 0 .. Show always<br><br>1 .. Show only when boarding<br><br>2 .. Show only when disembarking<br><br>4 .. Show only while moving<br><br>8 .. Show only when entering OR exiting the vehicle<br><br>16 .. Show only when entering AND exiting the vehicle | - |

*Table 42 Notice - Global Traffic Notice Texts*

If LINE_NR is filled, the notice is only valid for this line.

In order to correctly represent blanks, line breaks and special characters, the note texts in NOTICE_TEXT should be saved in apostrophes by export programmes. "\n" results in a new line.

### 5.6.5 notice_str.din

**DINO description (overview of all relations)**

Stop-related (or route-segment-related) information

| | Table: notice_str | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Range of values** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | line.version,<br><br>notice.version,<br><br>route.version,<br><br>stop_point.version<br><br>trip.version,<br><br>version.version |
| | Y | Optional | TIMETABLE_PERIOD | char (4) | | Timetable project | version.timetable_period |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | line.line_nr,<br><br>route.line_nr,<br><br>trip.line_nr |
| | N | Optional | STR_LINE_VAR | char (4) | | Route number | route.str_line_var |
| | N | Optional | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | route.line_dir_nr |

| | | | | | | |
|---|---|---|---|---|---|---|
| Y | Optional | TRIP_ID | decimal (8) | | Internal trip number | trip.trip_id |
| Y | Optional | LINE_CONSEC_NR | decimal (3) | | Consecutive stopping point number in the service journey pattern | route.line_consec_nr |
| N | Optional | STOP_NR | decimal (5) | 1..99999 | stop number | stop_point.stop_nr |
| N | Optional | STOPPING_POINT_NR | decimal (2) | | Stopping point number | stop_point.stopping_point_nr |
| Y (PK) | Obligation (mandatory) | HINW_STR_CODE | char (5) | | Key to the notice table | notice.hinw_str_code |

*Table 43 Notice_Str - Stop and route-related traffic information*

A range of tables are used to reflect the notices. The DINO format specifies the following notes

- Line-related notices
- Trip-related (timetable-related) notices
- Route-segment-related notices
- Stop-related notices

Notices are interpreted and imported into DIVA with the following logic:

## 5.6.5.1 Global line notice

Only the LINE_NR field needs to be filled for this. The note is set for all journeys during a DINO import to DIVA.

| **Table: notice_str - example line global notice** | | | | |
|---|---|---|---|---|
| **Key** | **Mandatory field** | **Attribute name** | **Comments** | **FK** |
| Y | Obligation (mandatory) | VERSION | Basic version | line.version, notice.version, |
| Y | Optional | TIMETABLE_PERIOD | Timetable project | version.timetable_period |
| Y | Obligation (mandatory) | LINE_NR | Internal line number | line.line_nr |
| N | ~~Optional~~ | STR_LINE_VAR | Remains empty | |
| N | ~~Optional~~ | LINE_DIR_NR | Remains empty | |
| Y | ~~Optional~~ | TRIP_ID | Remains empty | |
| Y | ~~Optional~~ | LINE_CONSEC_NR | Remains empty | |
| N | ~~Optional~~ | STOP_NR | Remains empty | |
| N | ~~Optional~~ | STOPPING_POINT_NR | Remains empty | |

| Table: notice_str - example line global notice | | | | |
|---|---|---|---|---|
| **Key** | **Mandatory field** | **Attribute name** | **Comments** | **FK** |
| Y (PK) | Obligation (mandatory) | HINW_STR_CODE | Key to the notice table | notice.hinw_str_code |

## 5.6.5.2 Trip-related notice

If the TRIP_ID is empty, the notice for a specific route of a line applies to all trips (on this route).

If the TRIP_ID, LINE_CONSEC_NR and STOP_NR fields are filled, the notice applies per trip and stop.

| Table: notice_str - Example journey-related notice | | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | line.version, notice.version, route.version, stop_point.version trip.version, version.version |
| | Y | Optional | TIMETABLE_PERIOD | char (4) | | Timetable project | version.timetable_period |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | line.line_nr, route.line_nr, trip.line_nr |
| | N | Optional | STR_LINE_VAR | char (4) | | Route number | route.str_line_var |
| | N | Optional | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | route.line_dir_nr |
| | Y | Optional | TRIP_ID | decimal (8) | | If empty, the note applies to all journeys on this route | trip.trip_id |
| | Y | Optional | LINE_CONSEC_NR | decimal (3) | | Consecutive stopping point number in the service journey pattern | route.line_consec_nr |
| | N | Optional | STOP_NR | decimal (5) | 1..99999 | Internal stop number | stop_point.stop_nr |
| | N | Optional | STOPPING_POINT_NR | decimal (2) | | Stopping point number | stop_point.stopping_point_nr |
| | Y (PK) | Obligation (mandatory) | HINW_STR_CODE | char (5) | | Key to the notice table | notice.hinw_str_code |

## 5.6.5.3 Route-segment-related notice

A route-segment-related notice applies to each route and stop. STR_LINE_VAR must be filled with LINE_CONSEC_NR.

| | Table: notice_str. - Example of a route-related note | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | notice.version, route.version, version.version |
| | Y | Optional | TIMETABLE_PERIOD | char (4) | | Timetable project | version.timetable_period |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | route.line_nr |
| | N | Optional | STR_LINE_VAR | char (4) | | Route number | route.str_line_var |
| | N | Optional | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | route.line_dir_nr |
| | Y | Optional | TRIP_ID | decimal (8) | | - | |
| | Y | Optional | LINE_CONSEC_NR | decimal (3) | | Consecutive stopping point number in the service journey pattern | route.line_consec_nr |
| | N | Optional | STOP_NR | decimal (5) | 1..99999 | Internal stop number | |
| | N | Optional | STOPPING_POINT_NR | decimal (2) | | - | |
| | Y (PK) | Obligation (mandatory) | HINW_STR_CODE | char (5) | | Key to the notice table | notice.hinw_str_code |

## 5.6.5.4 Stop-related notice

A stop-related notice can be implemented by entering a value in the STOP_NR field.

| | Table: notice_str. - Example of a stop-related notice | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | notice.version, stop_point.version, version.version |
| | Y | Optional | TIMETABLE_PERIOD | char (4) | | Timetable project | version.timetable_period |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | line.line_nr, |
| | N | Optional | STR_LINE_VAR | char (4) | | - | |
| | N | Optional | LINE_DIR_NR | decimal (3) | | - | |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | | | **Table: notice_str. - Example of a stop-related notice** | | | | |
| | Y | Optional | TRIP_ID | decimal (8) | | - | |
| | Y | Optional | LINE_CONSEC_NR | decimal (3) | | - | |
| | N | Optional | STOP_NR | decimal (5) | 1..99999 | Internal stop number | stop_point.stop_nr |
| | N | Optional | STOPPING_POINT_NR | decimal (2) | | Stopping point number | stop_point.stopping_point_nr |
| | Y (PK) | Obligation (mandatory) | HINW_STR_CODE | char (5) | | Key to the notice table | notice.hinw_str_code |

### 5.6.6    service_constraint.din

#### DINO description (overview of all relations)

List of trip and route-segment-related service constraints

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | | | **Table: service_constraint** | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | route.version, trip.version |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | routeline_nr, trip.line_nr |
| | N | Optional | STR_LINE_VAR | char (4) | | Route number | route.str_line_var |
| | N | Optional | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | route.line_dir_nr |
| | Y | Obligation (mandatory) | TRIP_ID | decimal (8) | | Internal trip number | trip.trip_id |
| | Y | Obligation (mandatory) | LINE_CONSEC_NR | decimal (3) | | Serial stopping point number in the service journey pattern | route.line_consec_nr |
| | N | Optional | STOP_NR | decimal (5) | 1..99999 | stop number | route.stop_nr |
| | N | Optional | STOPPING_POINT_NR | decimal (2) | | Stopping point number | route.stopping_point_nr |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | N | Obligation (mandatory) | SERVICE_INTERDICTION_CODE | char (1) | A, E,I<br><br>0 .. 9 | Type of service constraint:<br><br>A = Stops only for disembarking<br>E = Stops only for boarding<br>I = No service possible in intra-urban areas<br><br>0,1,2 ... depending on how many intra-urban area service constraints apply to the service journey pattern<br><br>B = Demand stop<br>M = Bicycle transport, stops only for alighting<br>N = Bicycle transport, Stops only for boarding<br>W = Bicycle transport, no inner-city transport possible<br>K = Operating stop: Course trolley<br>T = Operating stop<br>C = Demand stop, exit only<br>D = Demand stop, boarding only | - |

*Table 44 Service_Constraint - Journey and route-related service prohibitions*

**Intra-urban service constraints - SERVICE_INTERDICTION_CODE**

In addition to A, E and I, the SERVICE_INTERDICTION_CODE column can also contain the values 0, 1, 2, ..., depending on how many intra-urban service constraints apply to a service journey pattern.

You can see this in the following example:

```
VERSION;LINE_NR;STR_LINE_VAR;LINE_DIR_NR;TRIP_ID;LINE_CONSEC_NR;STOP_NR;ST
OPPING_POINT_NR;SERVICE_INTERDICTION_CODE;
 1;      27;    4;  1;   200028;  1; 1306;  6;I ;
 1;      27;    4;  1;   200028;  2; 9405;  1;0 ;
 1;      27;    4;  1;   200028;  3; 9410;  2;0 ;
 1;      27;    4;  1;   200028;  4; 9121;  1;0 ;
 1;      27;    4;  1;   200028;  5; 1305;  3;0 ;
 1;      27;    4;  1;   200028;  6; 8124;  2;1 ;
 1;      27;    4;  1;   200028;  7; 8123;  2;1 ;
 1;      27;    4;  1;   200028;  8;32146;  1;2 ;

 1;      27;    4;  1;   200029;  1; 1306;  6;I ;
 1;      27;    4;  1;   200029;  2; 9405;  1;0 ;
 1;      27;    4;  1;   200029;  3; 9410;  2;0 ;
 1;      27;    4;  1;   200029;  4; 9121;  1;0 ;
 1;      27;    4;  1;   200029;  5; 1305;  3;0 ;
 1;      27;    4;  1;   200029;  6; 8124;  2;1 ;
 1;      27;    4;  1;   200029;  7; 8123;  2;1 ;
 1;      27;    4;  1;   200029;  8;32146;  1;2 ;
```

Intra-urban service constraints are used to suppress the output of connections in the EFA router, e.g. where a faster regional bus serves an intra-urban route although there is a slower city bus. They can

be defined by marking a continuous part of a service journey pattern. The system does not check whether the stops on this route segment are located in the same municipality, for example.

The relevant DIVA data looks like this.



The two magenta-coloured areas indicate the service constraints. In DIVA, they are numbered consecutively starting with I1, then I2 and I3, etc. DINO counts the intra-urban service constraints starting with 0, i.e. in this case 0,1, and 2.

Specifically, the aim here was to prevent the EFA connections from issuing within Bad Herrenalb and within Bernbach. While it is possible to define I3 as a service constraint with one stop, it only serves as an example and makes no sense in this form.

Converted to DINO, the data looks like this:

```
DIVA-I1
1;     27;    4;   1;   200028;  1; 1306; 6;I ;
1;     27;    4;   1;   200028;  2; 9405; 1;0 ;
1;     27;    4;   1;   200028;  3; 9410; 2;0 ;
1;     27;    4;   1;   200028;  4; 9121; 1;0 ;
1;     27;    4;   1;   200028;  5; 1305; 3;0 ;


DIVA-I2
1;     27;    4;   1;   200028;  6; 8124; 2;1 ;
1;     27;    4;   1;   200028;  7; 8123; 2;1 ;


DIVA-I3
1;     27;    4;   1;   200028;  8;32146; 1;2 ;
```

The specification of the stop number is not mandatory, as the specification of LINE_CONSEC_NR is sufficient to identify the nth stop on the route. The DINO relation service_constraint.din refers directly to the route of the journey.

To ensure backward compatibility with DIVA 3, the intra-urban constraint is limited to values between 0 ... 9.
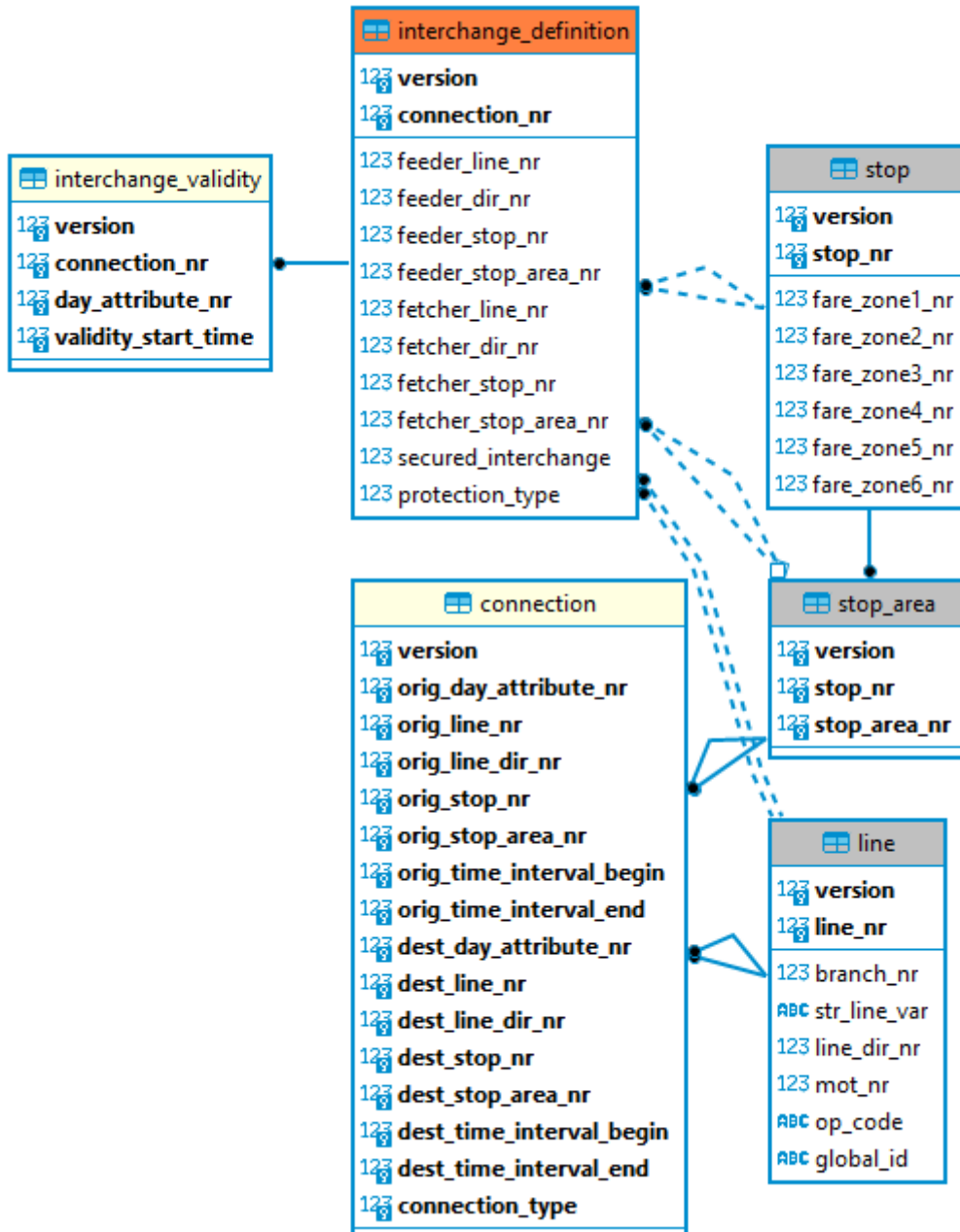
## 5.7  Connection data



*Figure 9 ER diagram connection data*

### 5.7.1  connection.din

This table is optional.

**DINO description ([overview of all relations](#))**

Through-service connections

The connections described in this relation are often referred to as through services. The important feature is that there is no change of vehicle for this kind of connection. In the case of railways, they can also be referred to as through trains. An example of a route connection is a bus that travels as line A to the main station, changes its destination sign there and continues as line B. An information system can use this to signal to passengers that they can remain seated in the vehicle in order to change from line A to line B (so-called "sitters").

| | | Table: connection | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | day_attribute.version, line.version, stop_area.version |
| | Y | Obligation (mandatory) | ORIG_DAY_ATTRIBUTE_NR | decimal (5) | | Day-type attribute number | day_attribute.orig_day_attribute_nr |
| | Y | Obligation (mandatory) | ORIG_LINE_NR | decimal (8) | | Departure line number | line.orig_line_nr |
| | Y | Obligation (mandatory) | ORIG_LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | line.orig_line_dir_nr |
| | Y | Obligation (mandatory) | ORIG_STOP_NR | decimal (5) | 1..99999 | Original internal stop number | stop_area.orig_stop_nr |
| | Y | Obligation (mandatory) | ORIG_STOP_AREA_NR | decimal (5) | 1..99998 | Number of the original stop area | stop_area.orig_stop_area_nr |
| | Y | Obligation (mandatory) | ORIG_TIME_INTERVAL_BEGIN | decimal (5) | 0..99999 | Start of transfer in seconds | - |
| | Y | Obligation (mandatory) | ORIG_TIME_INTERVAL_END | decimal (5) | 0..99999 | End of transfer in seconds | - |
| | Y | Obligation (mandatory) | DEST_DAY_ATTRIBUTE_NR | decimal (5) | | Day-type attribute number | day_attribute.dest_day_attribute_nr |
| | Y | Obligation (mandatory) | DEST_LINE_NR | decimal (8) | | Internal line number | line.dest_line_nr |
| | Y | Obligation (mandatory) | DEST_LINE_DIR_NR | decimal (3) | | Direction | line.dest_line_dir_nr |
| | Y | Obligation (mandatory) | DEST_STOP_NR | decimal (5) | 1..99999 | Destination stop number | stop_area.dest_stop_nr |
| | Y | Obligation (mandatory) | DEST_STOP_AREA_NR | decimal (5) | 1..99998 | Arrival stop area number | stop_area.dest_stop_area_nr |
| | Y | Obligation (mandatory) | DEST_TIME_INTERVAL_BEGIN | decimal (5) | 0..99999 | Start of transfer in seconds | - |
| | Y | Obligation (mandatory) | DEST_TIME_INTERVAL_END | decimal (5) | 0..99999 | End of transfer in seconds | - |
| | N | Obligation (mandatory) | TRANSFER_TIME | decimal (5) | 0..99999 | Transfer in seconds<br><br>Not applicable | - |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | | **Table: connection** | | | | | |
| | N | Optional | TRANSFER_DISTANCE | decimal (5) | 0..99999 | Transfer in meters<br><br>Not applicable | - |
| | Y | Obligation (mandatory) | CONNECTION_TYPE | decimal (1) | 1..9 | Type of through-service connection<br><br>1 = Passenger can remain seated in the vehicle,<br> 2 = Passenger can remain seated in the vehicle (intermediate destination is displayed),<br> 3 = Passenger can remain seated in the vehicle (final destination is displayed),<br> 4 = Through coach,<br> 5 = Change of category | - |

*Table 45 Connection - Optional journey bindings without vehicle change, seat stayers*

### 5.7.2 interchange_definition.din

This table is optional.

**DINO description (overview of all relations)**

Interchange definitions

Connection definitions are managed in a separate DINO relation. While connection.din contains through-service connection instructions without a change of vehicle, interchange definitions are defined in interchange_definition.din. These also include changing vehicles and waiting for vehicles within certain time windows. A distinction is made between secured (vehicles must wait for each other) and unsecured ("connection is usually reached") connection definitions.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | | **Table: interchange_definition** | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | line.version, stop_area.version, stop.version |
| | Y | Obligation (mandatory) | CONNECTION_NR | decimal (10) | | Unique number for an interchange definition, persistent | - |
| | N | Obligation (mandatory) | CONNECTION_NAME | Char (100) | ISO - 8859-1 | Freely definable text for identifying the interchange by name | - |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| | | | **Table: interchange_definition** | | | | |
| | N | Optional | CONTROL_CENTRE_CODE | Decimal (3) | 1..255(0) | Number of the control centre with which the interchange information is exchanged as per VDV 453. If the feeder service is within the area of the external control centre's responsibility, this attribute is assigned the value > 0. The value of this attribute controls the combination of attributes that is read in: If control centre identifier = 0, the following attributes are sent<br><br>• ZUB_LI_NR<br>• ZUB_LI_RI_NR<br>• ZUB_ORT_REF_ORT<br><br>If control centre identifier > 0<br><br>• LinienID,<br>• RichtungsID<br>• ASBID<br><br>. Attributes that are not supplied are filled with 0 or "". | - |
| | Y | Obligation (mandatory) | FEEDER_LINE_NR | decimal (8) | | Line number of the feeder service | line.line_nr |
| | Y | Obligation (mandatory) | FEEDER_DIR_NR | decimal (3) | 1..2(0) | Direction of the feeder line | line. dir_nr |
| | N | Obligation (mandatory) | FEEDER_STOP_NR | decimal (5) | 1..99999 | Stop at which passengers disembark from the feeder vehicle for the interchange. | stop_area. stop_nr, stop. stop_nr |
| | N | Optional | FEEDER_STOP_AREA_NR | decimal (5) | 1..99998 | Optional additional area information for FEEDER_STOP_NR | stop_area. stop_area_nr |
| | N | Optional | LINE_NR | Char(20) | ISO 8859-1 | Identifier of the feeder line; must be filled instead of the ZUB_LI_NR if feeder is in the area of responsibility of the third party service point. | - |
| | N | Optional | DIRECTION_NR | Char(20) | ISO 8859-1 | Identifier of the feeder line; must be filled instead of the ZUB_LI_RI_NR if the feeder is in the area of responsibility of the third-party service centre. | - |
| | N | Optional | CONNECTIONLINKREF | Char (20) | ISO 8859-1 | Interchange area ID. Must be reconciled with the interface partner and is only filled if it is an interchange to a third-party control centre. Number of a systematic interchange | - |
| | Y | Obligation (mandatory) | FETCHER_LINE_NR | decimal (8) | | Line number of the fetcher service | line.fetcher_line_nr |
| | Y | Obligation (mandatory) | FETCHER_DIR_NR | decimal (2) | 1..2(0) | Direction of the fetcher line | line.fetcher_dir_nr |
| | N | Obligation (mandatory) | FETCHER_STOP_NR | decimal (5) | 1..99999 | Stop at which passengers board the vehicle of the fetcher service | stop_area.fetcher_stop_nr,<br><br>stop.fetcher_stop_nr |

| | Table: interchange_definition | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | N | Optional | FETCHER_STOP_AREA_NR | decimal (5) | 1..99998 | Optional additional area information for FETCHER_STOP_NR | stop_area.fetcher_stop_area_nr |
| | N | Optional | SECURED_INTERCHANGE | decimal (1) | 1..99998 | Interchange synchronization<br><br>0 .. not synchronized<br><br>1 .. synchronized | - |
| | N | Optional | PROTECTION_TYPE | decimal (5) | \<empty\><br><br>1..99998 | Type of interchange<br><br>blank ... if SECURED_INTERCHANGE=0<br><br>otherwise<br><br>0 .. static<br><br>1 .. dynamic | - |

*Table 46 Interchange_Definition - Connection Definitions*

**Ex:**

```
VERSION;CONNECTION_NR;CONNECTION_NAME;CONTROL_CENTRE_CODE;FEEDER_LINE_NR;FEEDER_DIR_NR;FEEDER
_STOP_NR;FEEDER_STOP_AREA_NR;LINE_NR;DIRECTION_NR;CONNECTIONLINKREF;FETCHER_LINE_NR;FETCHER_D
IR_NR;FETCHER_STOP_AREA_NR;SECURED_INTERCHANGE;PROTECTION_TYPE;
21; 200;459-Altach Church 41-2b-s16-H -> 41-2b-s16-H 459-Altach Church ; 0; 345; 1; 459; ; ;
; ; 344; 1; 459; ;0; ;
21; 200;459-Altach Church 41-2b-s16-H -> 41-2b-s16-H 459-Altach Church ; 0; 345; 1; 459; ; ;
; ; 345; 1; 459; ;0; ;
21; 213;1574-Rankweil Mühlbach 15-61-s16-H -> 15-56-s16-H 1574-Rankweil Mühlbach ; 0; 193; 1;
1574; ; ; ; ; 182; 1; 1574; ;0; ;
21; 213;1574-Rankweil Mühlbach 15-61-s16-H -> 15-56-s16-H 1574-Rankweil Mühlbach ; 0; 193; 1;
1574; ; ; ; ; 183; 1; 1574; ;0; ;
21; 213;1574-Rankweil Mühlbach 15-61-s16-H -> 15-56-s16-H 1574-Rankweil Mühlbach ; 0; 193; 1;
1574; ; ; ; ; 184; 1; 1574; ;0; ;
```

### 5.7.1 Interchange_validity.din

This table is optional.

**DINO description (overview of all relations)**

Validity of connection definitions

Interchange monitoring can be restricted to a day type and to certain times of day. It is therefore possible to assign different validities to an interchange definition. Interchange monitoring may have different transfer and delay times depending on the time of day (description of transfer options or systematic interchanges).

| Table: interchange_validity | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | day_attribute.version, interchange_definition.version |
| | Y | Obligation (mandatory) | CONNECTION_NR | decimal (10) | 1.. | Unique number of the interchange definition, persistent | interchange_definition.connection_nr |
| | N | Optional | PRIORITY | Char (6) | ISO - 8859-1 | Free grouping of interchanges in terms of priority | - |
| | Y | Obligation (mandatory) | DAY_ATTRIBUTE_NR | decimal (3) | | Day type designation DAY_ATTRIBUTE_NR | day_attribute.day_attribute_nr |
| | Y | Obligation (mandatory) | VALIDITY_START_TIME | decimal (6) | 1.. | Time in seconds from midnight, from which the interchange definition is valid within the day type. | - |
| | N | Optional | VALIDITY_END_TIME | decimal (6) | 1.. | Time in seconds from midnight, until which the interchange definition is valid within the day type. | - |
| | N | Optional | INTERCHANGE_STANDARD_DURATION | decimal (6) | 1.. | Minimum transfer time for a transfer connection. Time in seconds available to a passenger to get from the stopping point of the feeder service to the stopping point of the fetcher service. | - |
| | N | Optional | INTERCHANGE_MAXIMUM_DURATION | decimal (6) | 1.. | Maximum interchange time for a transfer connection. Maximum time in seconds that the passenger can be expected to spend transferring between trains (including waiting time) that could still be referred to as an interchange. This attribute is used to form the interchange pairs. | - |
| | N | Optional | MAXIMUM_WAIT_TIME | decimal (6) | 1.. | Maximum timetable deviation in seconds which may be incurred by the fetcher if interchange synchronization is in effect | - |

| Table: interchange_validity | | | | | | |
|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | N | Optional | MAXIMUM_WAIT_TIME_AUTO | decimal (6) | 1.. | The system's margin of flexibility in seconds in the event of a deviation from the fetcher's service timetable. If this value is exceeded, a confirmation must be requested from the dispatcher for continued monitoring of this interchange. | - |

*Table 47 Interchange_Validity - Optional specification of restricted validity of connection definitions*

## 5.8   Route segments and georeferenced service journey patterns
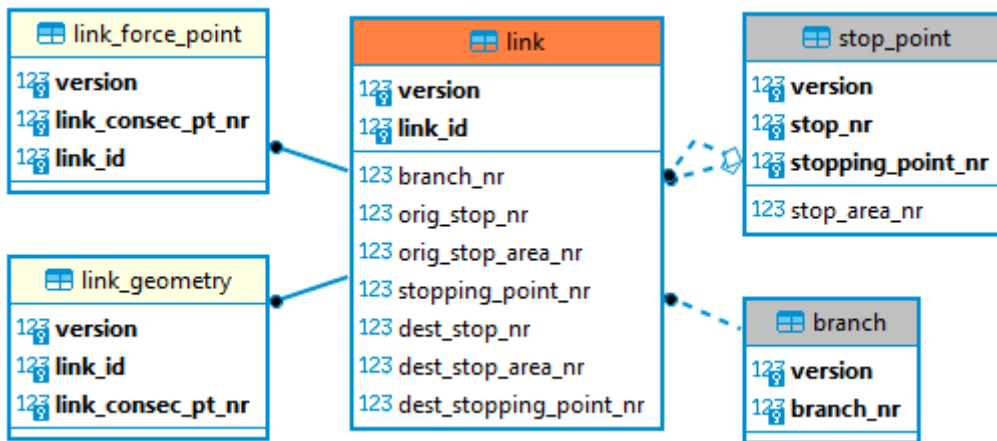


*Figure 10 ER diagram sub-routes and georeferenced routes*

### 5.8.1    link.din

This table is optional.

**DINO description (overview of all relations)**

Route segments

| Table: link | | | | | | |
|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | branch.version, stop_point.version |

| Table: link | | | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | LINK_ID | decimal (19) | 1.. 99999999 99999999 999 | Unique internal link ID, not persistent, used only for reference by child records | - |
| | Y | Obligation (mandatory) | BRANCH_NR | decimal (2) | 0..99 | Designation of the operating branch or authority | branch.branch_nr |
| | Y | Obligation (mandatory) | ORIG_STOP_NR | decimal (5) | 1..99999 | Original internal stop number | stop_point.orig_stop_nr |
| | Y | Optional | ORIG_STOP_AREA_NR | decimal (5) | 0..99998 | Number of the original stop area | - |
| | Y | Optional | STOPPING_POINT_NR | decimal (2) | | Stopping point number | stop_point.orig_stopping_point_nr |
| | Y | Obligation (mandatory) | DEST_STOP_NR | decimal (5) | 1..99999 | Destination stop number | stop_point.dest_stop_nr |
| | Y | Optional | DEST_STOP_AREA_NR | decimal (5) | 0..99998 | Number of the original stop area | - |
| | Y | Optional | DEST_STOPPING_POINT_NR | decimal (2) | | Stopping point number | stop_point.dest_stopping_point_nr |
| | N | Optional | LENGTH | decimal (8) | | Length in [m] | - |
| | N | Optional | GIS_LENGTH | decimal (8) | | GIS length in [m] | - |

*Table 48 Link - Optional table of track sections*

Route segments can be defined between platforms, areas and stops. The stopping point elements of the start and end points of the route segment may be of different types. Thus, one end point may be defined directly at the stop (or area "0") and the other at a riser.

For each operating branch, there may only be one unique route segment between two stop elements, e.g. there may only be one link between platform 1 of stop number 1 and platform 2 of stop number 2.

On the other hand, for another branch of operation, there may be further sub-routes for the same start and end points. For example, separate sections for buses and trams are possible between the same platforms.

### 5.8.2 link_geometry.din

This table is optional.

**DINO description (overview of all relations)**

Route segment polygons

If route segments that have already been georeferenced are to be transferred, the polygons of the referenced route segments can be transferred with the link_geometry relation. The route segment polygons should be optimized for transmission with a thinning algorithm (e.g. Douglas-Peucker) beforehand.

| | | Table: link_geometry | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandator y field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | link.version |
| | Y | Obligation (mandatory) | LINK_ID | decimal (19) | 1.. 9999999999999999999 | Unique internal non-persistent ID. Serves only as a reference to link.din | link.link_id |
| | Y | Obligation (mandatory) | LINK_CONSEC_PT_NR | decimal (5) | 1 .. 99999 | Sequential index in the route segment polygon | - |
| | N | Obligation (mandatory) | LINK_PT_X | decimal (11) | XXX.XXXXX | WGS84<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>-1 or blank entry means no coordinate<br><br>Other formats must be checked with MENTZ GmbH | - |
| | N | Obligation (mandatory) | LINK_PT_Y | decimal (11) | XXX.XXXXX | WGS84<br><br>If WGS84, then a value with up to 7 decimal places<br><br>16.1234567<br><br>Other formats must be checked with MENTZ GmbH | - |

*Table 49 Link_Geometry - Coordinate sequence of georeferenced partial routes*

### 5.8.3   link_force_point.din

This table is optional.

If, instead of transferring georeferenced route segments in link_geometry.din, they are to be georeferenced in the DIVA system, force points can be defined with the DINO link_force_point.din relation. DIVA can then use these force points for automated georeferencing.

**DINO description (overview of all relations)**

Force points

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| | | **Table: link_force_point** | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | link.version |
| | Y | Obligation (mandatory) | LINK_ID | decimal (19) | 1.. 9999999999999999999 | Unique internal non-persistent ID. Serves only as a reference to link.din | link.link_id |
| | Y | Obligation (mandatory) | LINK_CONSEC_PT_NR | decimal (2) | 1 .. 99 | Continuous index in route segment if there are several force points. | - |
| | N | Obligation (mandatory) | LINK_PT_X | decimal (11) | XXX.XXXXX | WGS84 If WGS84, then a value with up to 7 decimal places 16.1234567 -1 or blank entry means no coordinate Other formats must be checked with MENTZ GmbH | - |
| | N | Obligation (mandatory) | LINK_PT_Y | decimal (11) | XXX.XXXXX | WGS84 If WGS84, then a value with up to 7 decimal places 16.1234567 Other formats must be checked with MENTZ GmbH | - |

*Table 50 Link_Force_Point - constraint points for partial routes for later georeferencing*

## 5.9 User-defined attributes
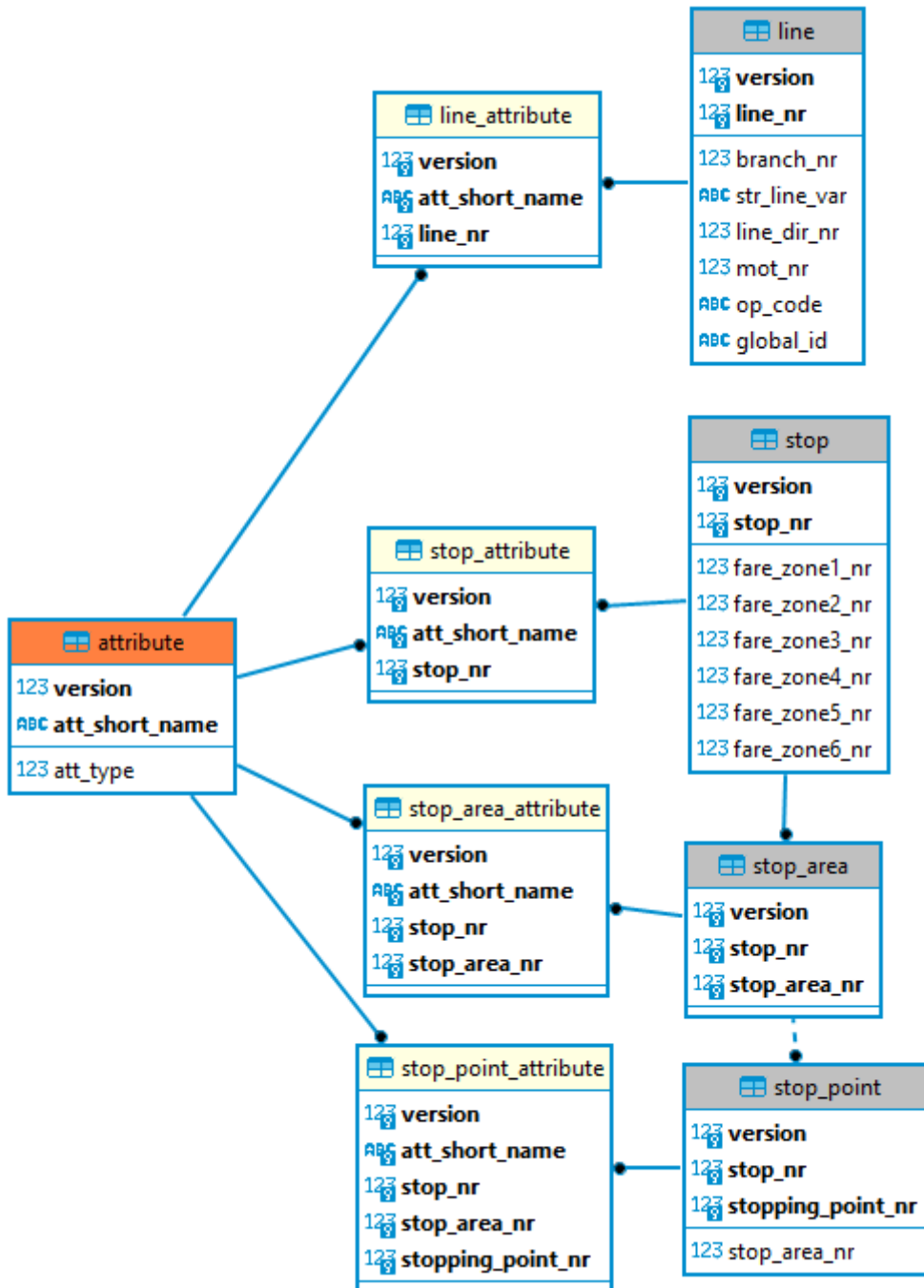


*Figure 11 ER Diagram Custom Attributes*

Diva 4 supports the creation of user-defined attributes. From version 2.0, the DINO interface also supports the definition of free numeric, Boolean, date, list and text attributes, which can be assigned to the relations stop, stop_area, stop_point and line and are transferred to DIVA 4 as network version-specific, user-defined attributes.

### 5.9.1 Attributes.din

This table is optional.

**DINO description (overview of all relations)**

Definition of the freely defined attributes

| | Table: attribute | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | version.version |
| | Y | Obligation (mandatory) | ATT_SHORT_NAME | Char(12) | | Unique, persistent short designation | - |
| | N | Obligation (mandatory) | ATT_LONG_NAME | Char(50) | | Description | - |
| | N | Obligation (mandatory) | ATT_TYPE | decimal(2) | 0 .. char<br><br>1 .. decimal<br><br>2 .. Date<br><br>3 .. list<br><br>4 .. bool | Attribute type | - |

*Table 51 Attributes - Freely definable attributes*

Remark: in older versions of the specification 2.1, the value range of the ATT_TYPE column was erroneously specified as (char | decimal | bool). The correct range of values is (0 | 1 | 2 | 3 | 4)

In the case of list attributes, the ATT_TYPE columns of the stop_attribute.din, stop_area_attribute.din, stop_point_attribute.din and line_attribute.din tables contain the short designation of the value exported from the field value list, i.e. usually the sequential number of the list value.

### 5.9.2 Stop_attribute.din

This table is optional.

| | Table: stop_attributes | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | attribute.version,<br><br>stop.version |
| | Y | Obligation (mandatory) | ATT_SHORT_NAME | char(12) | | Unique, persistent short designation | attribute.att_short_name |

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| **Table: stop_attributes** | | | | | | | |
| | N | Obligation (mandatory) | ATT_VALUE | Char(1000) | | value | - |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | For stops: stop number | stop.stop_nr |

*Table 52 Stop_Attribute - Assignment of optional stop attributes*

### 5.9.3  Stop_area_attribute.din

This table is optional.

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| **Table: stop_area_attributes** | | | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | attribute.version, stop_area.version |
| | Y | Obligation (mandatory) | ATT_SHORT_NAME | char(12) | | Unique, persistent short designation | attribute.att_short_name |
| | N | Obligation (mandatory) | ATT_VALUE | Char(1000) | | value | - |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | For stops: stop number | stop_area.stop_nr |
| | Y | Obligation (mandatory) | STOP_AREA_NR | decimal (5) | 1..99998 | Number of a stop area within a stop | stop_area.stop_area_nr |

*Table 53 Stop_Area_Attributes - Assignment of optional attributes to stop areas*

### 5.9.4  Stop_point_attribute.din

This table is optional.

<tbd> Description of what the purpose of this table is

| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
|---|-----|-----------------|----------------|-----------|-------------|----------|-----|
| **Table: stop_point_attributes** | | | | | | | |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | attribute.version, stop_point.version |

| | | | Table: stop_point_attributes | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | ATT_SHORT_NAME | char(12) | | Unique, persistent short designation | attribute.att_short_name |
| | N | Obligation (mandatory) | ATT_VALUE | Char(1000) | | value | - |
| | Y | Obligation (mandatory) | STOP_NR | decimal (5) | 1..99999 | For stops: stop number | stop_point.stop_nr |
| | Y | Obligation (mandatory) | STOP_AREA_NR | decimal (5) | 0..99998 | Number of a stop area within a stop | - |
| | Y | Obligation (mandatory) | STOPPING_POINT_NR | decimal (2) | 0..99 | Stopping point number | stop_point.stopping_point_nr |

*Table 54 Stop_Point_Attributes - Assignment of optional climb attributes*

### 5.9.5 line_attribute.din

This table is optional.

<tbd> Description of what the purpose of this table is

| | | | Table: line_attributes | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | 1..99 | Basic version | attribute.version, line.version |
| | Y | Obligation (mandatory) | ATT_SHORT_NAME | char(12) | | Unique, persistent short designation | attribute.att_short_name |
| | N | Obligation (mandatory) | ATT_VALUE | Char(1000) | | value | - |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Foreign key of the line. If set, then valid only for this line | line.line_nr |

*Table 55 Line_Attribute - Assignment of optional line attributes*

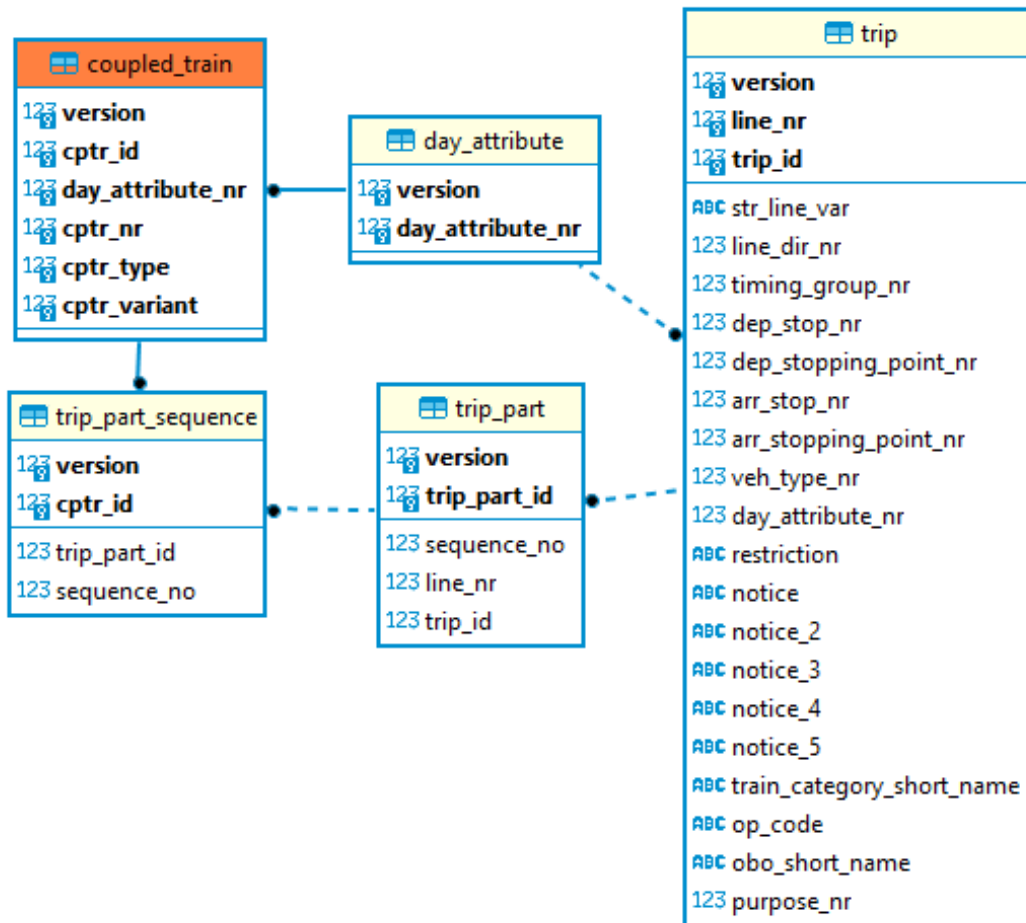## 5.10 Train scheduling: Definition of train sets (wing trains)



*Figure 12 ER diagram train planning and train compositions*

The following three tables describe the coupling of trips from trip.din to train sets.

All three tables together are optional. If one of the three tables is supplied, then all three must be supplied.

### 5.10.1 coupled_train.din

**DINO description (overview of all relations)**

convoy

| | Ke y | Mandator y field | Attribute name | Data type | Value range | Comments | FK |
|---|---|---|---|---|---|---|---|
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | day_attribute.version |

| | Table: coupled_train | | | | | |
|---|---|---|---|---|---|---|
| **T** | **Ke y** | **Mandator y field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | CPTR_ID | decimal (19) | 1.. 9999999999 999999999 | Unique internal non-persistent ID Used only to establish the reference of the self-reference from trip_part_sequence.d in | - |
| | Y | Obligation (mandatory) | DAY_ATTRIBUTE_NR | decimal (5) | | Day-type attribute number | day_attribute.day_attribute_nr |
| | Y | Obligation (mandatory) | CPTR_NR | decimal (10) | | Train composition number | - |
| | Y | Obligation (mandatory) | CPTR_TYPE | decimal (1) | | Type\n\n0 = Traffic train formation\n\n1 = Operational train formation\n\n2 = Traffic and operational train formation | - |
| | Y | Obligation (mandatory) | CPTR_VARIANT | decimal (2) | | Type | - |

*Table 56 Coupled_Train - Optional definition of a train set*

Various attributes of the train composition including the number and a DAY_ATTRIBUTE_NR which describes on which traffic days the coupling applies. These do not necessarily have to be the same traffic days as the coupled journeys. VERSION and CPTR_ID or VERSION, DAY_ATTRIBUTE_NR, CPTR_NR, CPTR_TYPE and CPTR_VARIANT form the key.

For example:

| VERSION | CPTR_ID | DAY_ATTRIBUTE_NR | CPTR_NR | CPTR_TYPE | CPTR_VARIANT |
|---|---|---|---|---|---|
| 1 | 26449635 | 47 | 2 | 2 | |
| 1 | 26449636 | 47 | 3 | 2 | |
| 1 | 26449634 | 564 | 1 | 2 | |
| 1 | 26449637 | 565 | 4 | 2 | |

### 5.10.2  trip_part.din

**DINO description (overview of all relations)**

Linking journeys

| | | Table: trip_part | | | | | |
|---|---|---|---|---|---|---|---|
| **T** | **Key** | **Mandatory field** | **Attribute name** | **Data type** | **Value range** | **Comments** | **FK** |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | trip.version |
| | Y | Obligation (mandatory) | TRIP_PART_ID | decimal (19) | 1.. 999999999 999999999 9 | Unique internal non-persistent ID. Serves only to establish the reference of the self-reference to trip_part.din | - |
| | N | Obligation (mandatory) | COUPLED_TRIP_PART_ID | decimal (19) | Empty, 1.. 999999999 999999999 9 | Empty for data set with TRAIN_POSITION = 1, otherwise reference to data set with TRAIN_POSITION = 1 of the train composition. | - |
| | Y | Obligation (mandatory) | SEQUENCE_NO | decimal (2) | 0..99 | Sequence number of the individual records of this table for a train composition, part of the key | - |
| | N | Obligation (mandatory) | TRAIN_POSITION | decimal (2) | 1..99 | Position of the train section in the whole train, must be unique for a train composition | - |
| | Y | Obligation (mandatory) | LINE_NR | decimal (8) | | Internal line number | trip.line_nr |
| | Y | Obligation (mandatory) | TRIP_ID | decimal (8) | | Internal trip number | trip.trip_id |
| | N | Optional | STR_LINE_VAR | char (4) | | Route number | - |
| | N | Optional | LINE_DIR_NR | decimal (3) | | Direction of travel, more than 2 directions are also possible | - |
| | N | Obligation (mandatory) | START_LINE_CONSEC_NR | decimal (3) | | Serial stopping point number in the service journey pattern | - |
| | N | Optional | START_STOP_NR | decimal (5) | 1..99999 | stop number | - |
| | N | Optional | START_STOPPING_POINT_ NR | decimal (2) | | Stopping point number | - |
| | N | Obligation (mandatory) | END_LINE_CONSEC_NR | decimal (3) | | Serial stopping point number in the service journey pattern | - |
| | N | Optional | END_STOP_NR | decimal (5) | 1..99999 | stop number | - |
| | N | Optional | END_STOPPING_POINT_NR | decimal (2) | | Stopping point number | - |

*Table 57 Trip_Part - Optional, definition of the connection of train parts*

COUPLED_TRIP_PART_ID represents the connection of the individual train parts to the main train part. The main train part has TRAIN_POSITION = 1 and no reference (COUPLED_TRIP_PART_ID = empty). The remaining columns identify the journey and start/finish in the route of the journey on which the train sections travel together. VERSION and TRIP_PART_ID form the key. VERSION, LINE_NR, TRIP_ID and TRAIN_POSITION (or SEQUENCE_NO) must be unique.

For example:

| VERSION | TRIP_PART_I | COUPLED_TRIP_PART_ID | SEQUENCE_NO | TRAIN_POSITION | LINE_NR | TRIP_ID | STR_LINE_VAR | LINE_DIR_NR | START_LINE_CONSEC_NR | START_STOP_NR | START_STOPPING_POINT_NR | END_LINE_CONSEC_NR | END_STOP_NR | END_STOPPING_POINT_NR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1141636 | | 1 | 1 | 1023 | 100089 | 8 | 1 | 2 | 168 | 26 | 7 | 55017 | 1 |
| 1 | 1141637 | 1141636 | 0 | 2 | 984 | 100007 | 3 | 2 | 1 | 168 | 26 | 6 | 55017 | 1 |
| 1 | 1141639 | | 0 | 1 | 1031 | 100007 | 5 | 2 | 1 | 2024 | 54 | 8 | 168 | 23 |
| 1 | 1141640 | 1141639 | 0 | 2 | 1043 | 100061 | 4 | 1 | 1 | 2024 | 54 | 8 | 168 | 23 |
| 1 | 1141643 | | 0 | 1 | 1023 | 100008 | 3 | 2 | 1 | 55017 | 1 | 7 | 168 | 23 |
| 1 | 1141644 | 1141643 | 0 | 2 | 1023 | 100055 | 5 | 2 | 1 | 55017 | 1 | 7 | 168 | 23 |
| 1 | 1141645 | 1141643 | 0 | 3 | 984 | 100047 | 9 | 1 | 1 | 55017 | 1 | 7 | 168 | 23 |
| 1 | 1141648 | | 1 | 1 | 1023 | 100090 | 8 | 1 | 2 | 168 | 26 | 3 | 179 | 27 |
| 1 | 1141649 | 1141648 | 1 | 2 | 1029 | 100024 | 7 | 1 | 6 | 168 | 26 | 7 | 179 | 27 |
| 1 | 1141650 | 1141648 | 0 | 3 | 984 | 100008 | 3 | 2 | 1 | 168 | 26 | 2 | 179 | 27 |

### 5.10.3  trip_part_sequence.din

**DINO description (overview of all relations)**

Connection of trip_part to coupled_train

| Table: trip_part_sequence | | | | | | | |
|---|---|---|---|---|---|---|---|
| T | Key | Mandatory field | Attribute name | Data type | Value range | Comments | FK |
| | Y | Obligation (mandatory) | VERSION | decimal (2) | | Basic version | trip_part.version, coupled_train.version |
| | Y | Obligation (mandatory) | CPTR_ID | decimal (19) | 1.. 9999999999999999999 | Unique internal non-persistent ID Used only to establish the reference to the record in the coupled_train table. | coupled_train.cptr_id |
| | Y | Obligation (mandatory) | TRIP_PART_ID | decimal (19) | 1.. 9999999999999999999 | Unique internal non-persistent ID Used only to establish the reference to the record in the trip_part table. | trip_part. trip_part_id |
| | Y | Obligation (mandatory) | SEQUENCE_NO | decimal (10) | 1..99 | The combination of coupled_train and sequence number must be unique. | - |

*Table 58 Trip_Part_Sequence - Assignment Trip_Part to Coupled_Train*

VERSION, CPTR_ID and SEQUENCE_NO or VERSION, CPTR_ID and TRIP_PART_ID form the key.

For example:

| VERSION | CPTR_ID | TRIP_PART_ID | SEQUENCE_NO |
|---|---|---|---|
| 1 | 26449634 | 1141636 | 1 |
| 1 | 26449635 | 1141639 | 1 |
| 1 | 26449636 | 1141643 | 1 |
| 1 | 26449637 | 1141648 | 1 |

TRIP_PART_ID only points to records in trip_part with empty COUPLED_TRIP_PART_ID and TRAIN_POSITION = 1

## 6    Annexes

### 6.1    Image directory

### 6.2    List of tables