

1 Allgemeines

Die Datenübergabe erfolgt zeilenweise in Textdateien (*.txt), die im ASCII-Format vorliegen.

Das erste Zeichen definiert die Bedeutung der Zeile.

- * Kommentarzeile
- \$ Relationsbeschreibung
(Anfang eines neuen Datenblocks mit dem Muster für die Daten,
die in diesem Block stehen;)

Alle nach der Relationsbeschreibung folgenden Zeilen bis zur nächsten Relationsbeschreibung bilden die Daten des Blocks. Die einzelnen Attribute sind durch das Trennzeichen ';' getrennt. Nach dem letzten Attribut einer Zeile steht kein Trennzeichen. Jede Zeile wird mit <carriage_return><line_feed> abgeschlossen.

2 Relationen - Fahrplandatei

2.1 Header

In der Relation „Header“ wird die Version der Schnittstellendatei beschrieben. Vorerst wird Version 1.0 verwendet. Bei jeder Änderung der Schnittstellendefinition muss eine neue Versionsnummer vereinbart werden.

\$VISION:VersNr;FileType

mit	VersNr	string
		Version der Schnittstelle
	FileType	string
		Typ der Schnittstellendatei

2.2 Haltestellen

In der Relation „Haltestellen“ werden die Daten für alle Haltestellen des Fahrplans hinterlegt (Start- und Endhaltestellen aller Fahrplanfahrten sowie Depots).

\$STOPPOINT:ID;Code;Name

mit	ID	int
		Eindeutiger Schlüssel für die Identifizierung der (End)Haltestelle
	Code	string
		kann beliebig gewählt werden
	Name	string
		kann beliebig gewählt werden

2.3 Linien

Die Relation „Linien“ definiert die im Fahrplan gefahrenen Linien. Jede Fahrplanfahrt ist genau einer Linie zugewiesen. Eine Linie kann mehrere Fahrplanfahrten enthalten.

\$LINE:ID;Code;Name

mit	ID	int
		Eindeutiger Schlüssel für die Identifizierung der Linie
	Code	string
		kann beliebig gewählt werden
	Name	string
		kann beliebig gewählt werden

2.4 Linienbündel

Die Relation „Linienbündel“ gruppiert Linien und definiert so, welche Linien zusammen geplant werden sollen. Jede Linie ist genau einem Linienbündel zugewiesen. Ein Linienbündel kann mehrere Linien enthalten.

\$LINEBUNDLE:ID;LineID

mit	ID	int
		Eindeutiger Schlüssel für die Identifizierung des Linienbündels
	LineID	int
		ID der Linie, die dem Linienbündel zugewiesen ist

2.5 Fahrzeugtypen

Die Relation „Fahrzeugtypen“ definiert die für die Fahrplanausführung einsetzbaren Fahrzeugtypen mit Kostenwerten und verfügbarer Anzahl.

\$VEHICLETYPE:ID;Code;Name;VehCost;KmCost;HourCost;Capacity

mit	ID	int
		Eindeutiger Schlüssel für die Identifizierung des Fahrzeugtyps
	Code	string
		kann beliebig gewählt werden
	Name	string
		kann beliebig gewählt werden
	VehCost	float
		Kosten für den Einsatz eines Fahrzeugs dieses Fahrzeugtyps (z.B. Abschreibungskosten für den Planungszeitraum)
	KmCost	float
		Kosten, die pro Kilometer Fahrleistung eines Fahrzeugs dieses Typs anfallen
	HourCost	float
		Kosten, die pro Stunde Dienst eines Fahrzeugs dieses Typs anfallen (auch Wartezeiten außerhalb des Depots)
	Capacity	int
		Anzahl maximal einsetzbarer Fahrzeuge dieses Typs

2.6 Fahrzeugtypgruppen

Die Relation „Fahrzeugtypgruppen“ definiert Fahrzeugtypgruppen. Jede Fahrplanfahrt ist einer Fahrzeugtypgruppe zugewiesen (s.u.). Die Fahrplanfahrt kann von jedem Fahrzeugtyp ihrer Gruppe bedient werden.

\$VEHICLETYPEGROUP:ID;Code;Name

mit	ID	int	Eindeutiger Schlüssel für die Identifizierung der Fahrzeugtypgruppe
	Code	string	kann beliebig gewählt werden
	Name	string	kann beliebig gewählt werden

2.7 FahrzeugtypZuFahrzeugtypgruppe

Die Relation „FahrzeugtypZuFahrzeugtypgruppe“ weist Fahrzeugtypen zu Fahrzeugtypgruppen zu und definiert so, welche Fahrzeugtypen zusammen geplant werden können. Ein Fahrzeugtyp kann mehreren Fahrzeugtypgruppen zugewiesen werden.

\$VEHTYPETOVEHTYPEGROUP:VehTypeID;VehTypeGroupID

mit	VehTypeID	int	ID des zuzuweisenden Fahrzeugtyps
	VehTypeGroupID	int	ID der Fahrzeugtypgruppe, zu der der Fahrzeugtyp zugewiesen wird

2.8 FahrzeugkapazitätenZuHaltestelle

Die Relation „FahrzeugkapazitätenZuHaltestelle“ bestimmt die (Stellplatz-)Kapazitäten für einen Fahrzeugtyp eines Depots.

\$VEHTYPECAPTOSTOPPOINT:VehTypeID;StoppointID;Min;Max

mit	VehTypeID	int	ID des Fahrzeugtyps, für den die Kapazität festgelegt wird
	StoppointID	int	ID der Depot-Haltestelle, für die die Kapazität festgelegt wird
	Min	int	Anzahl minimal einzusetzender Fahrzeuge dieses Typs aus diesem Depot
	Max	int	Anzahl maximal einsetzbarer Fahrzeuge dieses Typs aus diesem Depot

2.9 Linienfahrten

Die Relation „Linienfahrten“ definiert die zu verplanenden Fahrgastfahrten des Fahrplans.

\$SERVICEJOURNEY:ID;LineID;FromStopID;ToStopID;DepTime;ArrTime;MinAheadTime;MinLayoverTime;VehTypeGroupID;MaxShiftBackwardSeconds;MaxShiftForwardSeconds;FromStopBreakFacility;ToStopBreakFacility;CODE

mit	ID	int	Eindeutiger Schlüssel für die Identifizierung der Fahrplanfahrt
	LineID	int	ID der Linie, in welcher diese Fahrt bedient wird
	FromStopID	int	ID der Haltestelle, von der die Fahrplanfahrt startet
	ToStopID	int	ID der Haltestelle, an der die Fahrplanfahrt endet
	DepTime	DDD:HH:MM:SS	Abfahrtszeit der Fahrplanfahrt
	ArrTime	DDD:HH:MM:SS	Ankunftszeit der Fahrplanfahrt
	MinAheadTime	int	Minimale Vorbereitungszeit vor Beginn der Fahrt in Sekunden
	MinLayoverTime	int	Minimale Wendezeit nach Ende der Fahrt in Sekunden
	VehTypeGroupID	int	ID der Fahrzeugtypgruppe, von dessen Fahrzeugtypen diese Fahrt bedient werden kann
	MaxShiftBackwardSeconds	int	Maximal mögliche frühere Bedienung der Fahrplanfahrt
	MaxShiftForwardSeconds	int	Maximal mögliche spätere Bedienung der Fahrplanfahrt
	FromStopBreakFacility	bool (Wertebereich {0;1} mit 0 = Nein und 1 = Ja)	Gibt an, ob vor Beginn der Fahrt eine Pausenmöglichkeit besteht
	ToStopBreakFacility	bool (Wertebereich {0;1} mit 0 = Nein und 1 = Ja)	Gibt an, ob nach Beendigung der Fahrt eine Pausenmöglichkeit besteht
	Code	string	Eindeutige Kennung der Fahrgastfahrt

2.10 Verbindungsfahrten

Die Relation „Verbindungsfahrten“ definiert tageszeitabhängige Verbindungsdaten für die Durchführung von Leerfahrten zwischen zwei Haltestellen.

\$DEADRUNTIME:FromStopID;ToStopID;FromTime;ToTime;Distance;RunTime

mit	FromStopID	int	ID der Starthaltestelle der Verbindungsangabe
	ToStopID	int	ID der Zielhaltestelle der Verbindungsangabe
	FromTime	DDD:HH:MM:SS	

ToTime	Anfangszeit für den Zeitraum, in welchem diese Verbindung gültig ist DDD:HH:MM:SS
Distance	Endzeit für den Zeitraum, in welchem diese Verbindung gültig ist int Wegstrecke einer Verbindungsfahrt zwischen Start- und Zielhaltestelle in Metern
RunTime	int Dauer einer Verbindungsfahrt zwischen Start- und Zielhaltestelle in Sekunden

2.11 Ablösepunkte

Die Relation „Ablösepunkte“ definiert durch Haltestelle und Zeitpunkt definierte Möglichkeiten, vor/während/nach einer Fahrplanfahrt abzulösen.

\$RELIEFPOINT:ID;ServicejourneyID;StoppointID;Arrivaltime

mit	ID	int Eindeutige Kennzeichnung des Ablösepunktes
	ServicejourneyID	int ID der Fahrplanfahrt, auf welcher diese Ablösemöglichkeit besteht
	StoppointID	int ID der Haltestelle, an welcher für die Fahrplanfahrt zum gegebenen Zeitpunkt eine Ablösemöglichkeit besteht
	Arrivaltime	DDD:HH:MM:SS Zeitpunkt, zu welchem abgelöst werden kann

2.12 Transferzeiten

Die Relation „Transferzeiten“ definiert die Verbindungszeit für den Transfer eines Fahrers zwischen zwei Haltestellen.

\$TRANSFERTIME:FromStopID;ToStopID;Duration

mit	FromStopID	int ID der Starthaltestelle des Transfers
	ToStopID	int ID der Zielhaltestelle des Transfers
	Duration	int Dauer des Transfers in Sekunden

3 Relationen - Dienstregeldatei

3.1 Dienstregeln

Die Relation „\$DUTYTYPE“ definiert Diensttypen u.a. über Schichtangaben, Pausentypen, Angaben zur Arbeitszeit, Dauer und Lenkzeit, sowie zu Kostensätzen.

\$DUTYTYPE:name;startTimeMin;startTimeMax;endTimeMin;endTimeMax;signOnTime;signOffTime;durationMin;durationMax;workingTimeTotalMin;workingTimeTotalMax;workingTimeBeforeFirstBreakMin;workingTimeWithoutBreakMin;workingTimeAfterLastBreakMin;drivingTimeTotalMin;drivingTimeTotalMax;drivingTimeWithoutBreakMin;drivingTimeWithoutBreakMax;drivingTimeWithoutBreakMinInterruptionTime;drivingTimeBeforeFirstBreakMin;breakType;breakTimeTotalMin;breakTimeTotalMax;breakTimeMin;breakTimeMax;pieceCountMin;pieceCountMax;allowedCumulatedWaitingTimeMax;dutyFixCost;isWorkRateConsidered;isBreakRateConsidered;dutyCostPerMinute;isVehicleChangeAllowedDuringBreak;breakTimeAllowedStarts;breakTimeAllowedEnds

mit	name	string
		kann beliebig gewählt werden
	startTimeMin	DDD:HH:MM:SS Frühester Anfangszeitpunkt
	startTimeMax	DDD:HH:MM:SS Spätester Anfangszeitpunkt
	endTimeMin	DDD:HH:MM:SS Frühester Endzeitpunkt
	endTimeMax	DDD:HH:MM:SS Spätester Endzeitpunkt
	signOnTime	DDD:HH:MM:SS Vorbereitungszeit
	signOffTime	DDD:HH:MM:SS Nachbereitungszeit
	durationMin	DDD:HH:MM:SS Minimale Gesamtdauer
	durationMax	DDD:HH:MM:SS Maximale Gesamtdauer
	workingTimeTotalMin	DDD:HH:MM:SS Minimale Gesamtarbeitszeit
	workingTimeTotalMax	DDD:HH:MM:SS Maximale Gesamtarbeitszeit
	workingTimeBeforeBreakMin	DDD:HH:MM:SS Minimale Arbeitszeit vor der ersten Pause
	workingTimeWithoutBreakMin	DDD:HH:MM:SS Minimale Arbeitszeit ohne Pausenunterbrechung
	workingTimeAfterLastBreakMin	DDD:HH:MM:SS Minimale Arbeitszeit nach der letzten Pause
	drivingTimeTotalMin	DDD:HH:MM:SS Minimale Gesamtlenkzeit
	drivingTimeTotalMax	DDD:HH:MM:SS

	Maximale Gesamtlenkzeit
drivingTimeWithoutBreakMin	DDD:HH:MM:SS Minimale Lenkzeit ohne (Pausen)Unterbrechung
drivingTimeWithoutBreakMax	DDD:HH:MM:SS Maximale Lenkzeit ohne (Pausen)Unterbrechung
drivingTimeWithoutBreakMinInterruptionTime	DDD:HH:MM:SS Minimale Zeit ab welcher eine Unterbrechung der Lenkzeit zum Zurücksetzen der aktuell kumulierten Lenkzeit führt
drivingTimeBeforeFirstBreakMin	DDD:HH:MM:SS Minimale Lenkzeit vor der ersten Pause
breakType	string (Wertebereich = {Tripper, Split, Block15, Block20, Block30, Ratio_1_5, Ratio_1_6}) Art des Pausentypes mit Tripper = keine Pause Split = Dienstteilung Block15 = mindestens 3 Block-Pausen Block20 = mindestens 2 Block-Pausen Block30 = mindestens 1 Block-Pause Ratio_1_5 = Verhältnispause 1/5 Ratio_1_6 = Verhältnispause 1/6
breakTimeTotalMin	DDD:HH:MM:SS Minimale Gesamtpausenzeit
breakTimeTotalMax	DDD:HH:MM:SS Maximale Gesamtpausenzeit
breakTimeMin	DDD:HH:MM:SS Minimale Pausenzeit
breakTimeMax	DDD:HH:MM:SS Maximale Pausenzeit
pieceCountMin	int Minimale Anzahl Dienststücke
pieceCountMax	int Maximale Anzahl Dienststücke
allowedCumulatedWaitingTimeMax	DDD:HH:MM:SS Maximal erlaubte Zeit die an einer Haltestelle gewartet werden darf, an der keine Pausenmöglichkeit besteht
dutyFixCost	float Fixkosten
isWorkRateConsidered	bool (Wertebereich {0;1} mit 0 = Nein und 1 = Ja) Gibt an, ob für die Arbeitszeit Kosten anfallen
isBreakRateConsidered	bool (Wertebereich {0;1} mit 0 = Nein und 1 = Ja) Gibt an, ob für die Pausenzeit Kosten anfallen
dutyCostPerMinute	float Kosten, die pro Minute anfallen
isVehicleChangeAllowedDuringBreak	bool (Wertebereich {0;1} mit 0 = Nein und 1 = Ja) Gibt an, ob in/vor/nach der Pause das Fahrzeug gewechselt werden darf

breakTimeAllowedStarts	List<DDD:HH:MM:SS,> Auflistung der erlaubten Startzeitpunkte von Zeiträumen, in denen eine Pause liegen darf; (der an der entsprechenden Position in „breakTimeAllowedEnds“ geschriebene Wert definiert den Endzeitpunkt des Pausenzeitraumes)
breakTimeAllowedEnds	List<DDD:HH:MM:SS,> Auflistung der erlaubten Endzeitpunkte von Zeiträumen, in denen eine Pause liegen darf; (der an der entsprechenden Position in „breakTimeAllowedStarts“ geschriebene Wert definiert den Startzeitpunkt des Pausenzeitraumes)

4 Relationen - Umlaufplandatei

4.1 Umläufe

In der Relation „Umläufe“ werden die benötigten Umläufe mit ihrer Zuordnung zu einem Depot sowie Fahrzeugtyp definiert.

\$BLOCK:ID;VehTypeID;DepotID

mit	ID	int	Eindeutige Kennzeichnung des Umlaufs
	VehTypeID	int	ID des verwendeten Fahrzeugtyps
	DepotID	int	ID des zugeordneten Depots

4.2 FahrtZuUmlauf

In der Relation „FahrtZuUmlauf“ werden jedem Umlauf die auszuführenden Fahrten zugewiesen und definiert. Wartezeiten an Haltestellen werden nicht explizit angegeben, sondern ergeben sich aus den Zeiten, die zwischen zwei aufeinanderfolgenden Fahrten liegen.

\$BLOCKELEMENT:BlockID;ServiceJourneyID;FromStopID;ToStopID;DepTime;ArrTime;ElementType;ServiceJourneyCode

mit	BlockID	int	ID des Umlaufs zu welchem diese Fahrt zugeordnet ist
	ServiceJourneyID	string	ID der Fahrt
	FromStopID	int	ID der Abfahrtshaltestelle
	ToStopID	int	ID der Ankunftshaltestelle
	DepTime	DDD:HH:MM:SS	Zeitpunkt der Abfahrt
	ArrTime	DDD:HH:MM:SS	Zeitpunkt der Ankunft
	ElementType	int (Wertebereich = {1,2,3,4,5,6,9,10})	Art der Fahrt
		mit	1 = Fahrplanfahrt 10 = LayoverTime
			2 = Verbindungsfahrt
			3 = Einrückfahrt in das Depot
			4 = Ausrückfahrt aus dem Depot
			5 = Fahrzeugvorbereitung
			6 = Fahrzeugnachbereitung
			9 = Warten
	ServiceJourneyCode	string	Code der Fahrplanfahrt

5 Relationen - Dienstplandatei

5.1 Dienste

In der Relation „Dienste“ werden die benötigten Dienste mit ihrem Typ definiert.

\$DUTY:ID;DutyType

mit	ID	int	
			Eindeutige Kennzeichnung des Dienstes
	DutyType	string	
			Beschreibung des Diensttyps

5.2 Dienstelemente

In der Relation „Dienstelemente“ werden jedem Dienst die auszuführenden Aufgaben zugewiesen und definiert. Wartezeiten an Haltestellen werden nicht explizit angegeben, sondern ergeben sich aus den Zeiten, die zwischen zwei aufeinanderfolgenden Aufgaben liegen.

\$DUTYELEMENT:DutyID;BlockID;ServiceJourneyID;FromStopID;ToStopID;DepTime;ArrTime;ElementType; ServiceJourneyCode

mit	DutyID	int	
			ID des Dienstes zu welchem diese Aufgabe zugeordnet ist
	BlockID	string	
			ID des Umlaufs welcher diese Aufgabe enthält
	ServiceJourneyID	string	
			ID der Fahrt
	FromStopID	string	
			ID der Haltestelle, an welcher die Aufgabe beginnt
	ToStopID	string	
			ID der Haltestelle, an welcher die Aufgabe endet
	DepTime	DDD:HH:MM:SS	
			Zeitpunkt des Aufgabenbeginns
	ArrTime	DDD:HH:MM:SS	
			Zeitpunkt des Aufgabendes
	ElementType	int (Wertebereich = {1,2,3,4,5,6,7,8,9,10})	Art der Aufgabe
		mit	1 = Fahrplanfahrt 10 = LayoverTime
			2 = Verbindungsfahrt
			3 = Einrückfahrt in das Depot
			4 = Ausrückfahrt aus dem Depot
			5 = Vorbereitung
			6 = Nachbereitung
			7 = Transfer
			8 = Pause
			9 = Warten
	ServiceJourneyCode	string	
			Code der Fahrplanfahrt