

SS2010
BAI2-DBP Gruppe 1
Lösung zu Übungsblatt 4

R. C. Ladiges, D. Fast

19. Mai 2010

Inhaltsverzeichnis

9 Aufgabe 9 (Views und Rechte)	5
9.0 Aufgabenstellung	5
9.1 SQL-Befehle	5
10 Aufgabe 10 (UML- und ER-Modell)	6
10.0 Aufgabenstellung	6
10.1 A23	6
10.1.1 Anforderung	6
10.1.2 UML-Diagramm	6
10.2 A24	6
10.2.1 Anforderung	6
10.2.2 UML-Diagramm	6
10.3 A25	7
10.3.1 Anforderung	7
10.3.2 UML-Diagramm	7
10.3.3 Anmerkungen	7
10.4 A26	8
10.4.1 Anforderung	8
10.4.2 UML-Diagramm	8
10.4.3 Anmerkungen	8
10.5 A27	8
10.5.1 Anforderung	8
10.5.2 UML-Diagramm	8
10.6 A28	9
10.6.1 Anforderung	9
10.6.2 UML-Diagramm	9
10.7 A29	9
10.7.1 Anforderung	9
10.7.2 UML-Diagramm	9
10.8 A30	10
10.8.1 Anforderung	10
10.8.2 UML-Diagramm	10
10.9 A31	10
10.9.1 Anforderung	10
10.9.2 Anmerkungen	10
10.10 A32	10
10.10.1 Anforderung	10
10.10.2 UML-Diagramm	10
10.11 A33	11
10.11.1 Anforderung	11
10.11.2 UML-Diagramm	11
10.11.3 Anmerkungen	11
10.12 A34	11
10.12.1 Anforderung	11
10.12.2 UML-Diagramm	11
10.12.3 Anmerkungen	11
10.13 A35	12

10.13.1 Anforderung	12
10.13.2 UML-Diagramm	12
10.14A36	12
10.14.1 Anforderung	12
10.14.2 UML-Diagramm	12
10.15A37	13
10.15.1 Anforderung	13
10.15.2 UML-Diagramm	13
10.16A38	13
10.16.1 Anforderung	13
10.16.2 UML-Diagramm	13
10.17A39	14
10.17.1 Anforderung	14
10.17.2 UML-Diagramm	14
10.17.3 Anmerkungen	14
10.18A40	14
10.18.1 Anforderung	14
10.18.2 Anmerkungen	14
10.19A41	15
10.19.1 Anforderung	15
10.19.2 UML-Diagramm	15
10.20A42	15
10.20.1 Anforderung	15
10.20.2 UML-Diagramm	15
10.21A43	16
10.21.1 Anforderung	16
10.21.2 UML-Diagramm	16
10.22A44	17
10.22.1 Anforderung	17
10.22.2 UML-Diagramm	17
10.23A45	17
10.23.1 Anforderung	17
10.23.2 Anmerkungen	17
10.24A46	18
10.24.1 Anforderung	18
10.24.2 UML-Diagramm	18
10.25A47	19
10.25.1 Anforderung	19
10.25.2 UML-Diagramm	19
10.26A48	19
10.26.1 Anforderung	19
10.26.2 Anmerkungen	19
10.27A49	20
10.27.1 Anforderung	20
10.27.2 UML-Diagramm	20
10.28A50	21
10.28.1 Anforderung	21
10.28.2 UML-Diagramm	21
10.28.3 Anmerkungen	21

10.29A51	22
10.29.1 Anforderung	22
10.29.2 UML-Diagramm	22
10.30SQL-Skript	22
11 Aufgabe 11 (SQL)	27
11.0 Aufgabenstellung	27

9 Aufgabe 9 (Views und Rechte)

9.0 Aufgabenstellung

In unserem HLS-Szenario stellt jede Zweiergruppe in der Praktikumsgruppe einen potenziellen Geschäftspartner dar. Organisieren Sie Ihre Datenbank so, dass nur Sie selbst und der jeweilige Geschäftspartner lesenden und schreibenden Zugriff auf die Daten des Partners bekommen. Testen Sie dies innerhalb der Praktikumsgruppe aus (mit Hilfe des Oracle SQL Developers)!

9.1 SQL-Befehle

```
create role GPRolle;

grant SELECT, INSERT on db.Geschaeftpartner to GPRolle;
grant SELECT, INSERT on db.GP_Typ to GPRolle;
grant SELECT, INSERT on db.Adresse to GPRolle;
grant SELECT, INSERT on db.Ziel to GPRolle;
grant SELECT, INSERT on db.Quelle to GPRolle;
grant SELECT, INSERT on db.TPZ_con_LOK to GPRolle;
grant SELECT, INSERT on db.TPZ_in_TPN to GPRolle;
grant SELECT, INSERT on db.Transportabschnitt to GPRolle;
grant SELECT, INSERT on db.Transportzonen to GPRolle;
grant SELECT, INSERT on db.Zonen_Typ to GPRolle;
grant SELECT, INSERT on db.PLZ_Zone to GPRolle;
grant SELECT, INSERT on db.Direkte_Zone to GPRolle;
grant SELECT, INSERT on db.LOK_in_TPN to GPRolle;
grant SELECT, INSERT on db.Transportnetzwerk to GPRolle;
grant SELECT, INSERT on db.Lokationen to GPRolle;
grant SELECT, INSERT on db.Lok_Art to GPRolle;
grant SELECT, INSERT on db.Kundenrechnungen to GPRolle;
grant SELECT, INSERT on db.Kundenfrachtabrechnungen to GPRolle;
grant SELECT, INSERT on db.Waren to GPRolle;
grant SELECT, INSERT on db.Sendungspositionen to GPRolle;
grant SELECT, INSERT on db.Sendungsanfragen to GPRolle;
grant SELECT, INSERT on db.Verkehrszweig to GPRolle;
grant SELECT, INSERT on db.Fahrzeug to GPRolle;
grant SELECT, INSERT on db.Transportmittel to GPRolle;
grant SELECT, INSERT on db.TP_M_Bez to GPRolle;
grant SELECT, INSERT on db.Transportbeziehung to GPRolle;
grant SELECT, INSERT on db.TPA_Transport to GPRolle;
grant SELECT, INSERT on db.TPA_Entladen to GPRolle;
grant SELECT, INSERT on db.TPA_Laden to GPRolle;
grant SELECT, INSERT on db.Kapazitaet to GPRolle;
grant SELECT, INSERT on db.Transportaktivitaet to GPRolle;
grant SELECT, INSERT on db.Transportplan to GPRolle;
grant SELECT, INSERT on db.Frachteinheit to GPRolle;

grant GPRolle to aaw688;
grant GPRolle to aaw891;
```

10 Aufgabe 10 (UML- und ER-Modell)

10.0 Aufgabenstellung

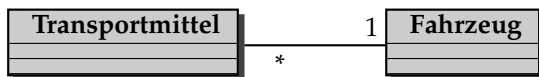
Modellieren Sie die Anforderungen [A23] bis [A51] des HAW Logistics-Lastenheftes in Form von UML-Klassendiagrammen. Ergänzen Sie Ihre Datenbank um die entstandenen Tabellen. Abzugeben ist ein PDF-Dokument mit den UML-Modellen dem SQL-Skript, das die gesamte DB (inkl. der vorhergehenden Tabellen) realisiert.

10.1 A23

10.1.1 Anforderung

Ein Transportmittel repräsentiert eine Klasse von Fahrzeugen [A28].

10.1.2 UML-Diagramm

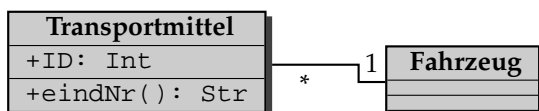


10.2 A24

10.2.1 Anforderung

Transportmittel sind durch eine Nummer der Form „TM-n“ (n ist eine natürliche Zahl) eindeutig identifizierbar.

10.2.2 UML-Diagramm



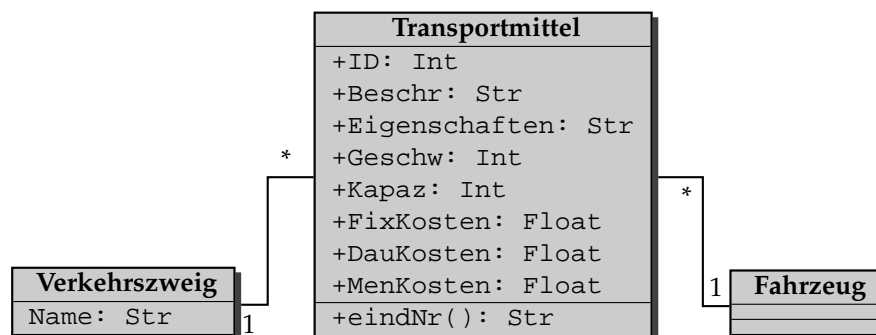
10.3 A25

10.3.1 Anforderung

Jedem Transportmittel ist einem Verkehrszweig (See, Luft, Land) zugeordnet. Außerdem ist es durch folgende Attribute beschrieben:

- Beschreibung, z. B. „LKW“, „Großes Schiff“, „Frachtflugzeug/CAO“
- mehrere Eigenschaften, dies können beliebige textuelle Angaben sein.
- Niedrige, mittlere und hohe Geschwindigkeit zur Berechnung der Kosten und der Dauer
- Einer Kapazität in FEU oder TEU
- Fixkosten pro Fahrt, Entfernungskosten (pro km Benutzung), Dauerkosten (pro Minute Benutzung) und Mengenkosten (pro transportiertem TEU/FEU)

10.3.2 UML-Diagramm



10.3.3 Anmerkungen

`+Geschw: Int`, 1=Langsam, 2=Mittel und 3=Schnell. Besser: Physikalische Größe in bestimmter Einheit.

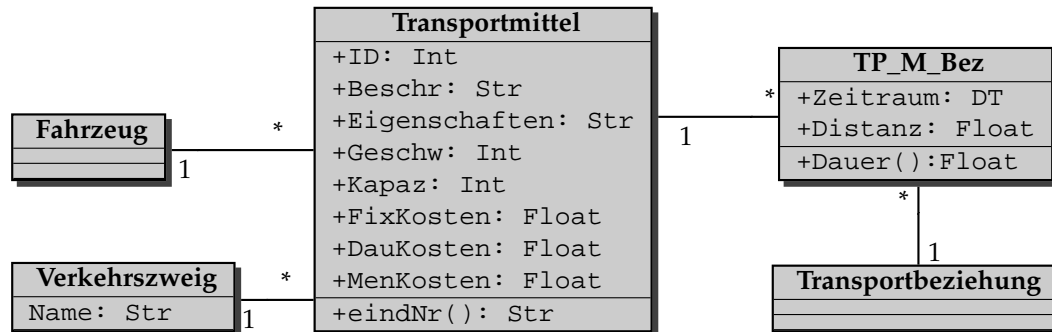
`+Kapaz: Int`, Angabe in TEU. 1 FEU = 2 TEU.

10.4 A26

10.4.1 Anforderung

Ein Transportmittel kann auf mehreren Transportbeziehungen [A10] eingesetzt werden. Die Verwendung eines Transportmittels auf einer Transportbeziehung ist durch einen Gültigkeitszeitraum, der Distanz sowie der Dauer bestimmt. Die Dauer ist aus Distanz und mittlerer Geschwindigkeit des Transportmittels abzuleiten.

10.4.2 UML-Diagramm



10.4.3 Anmerkungen

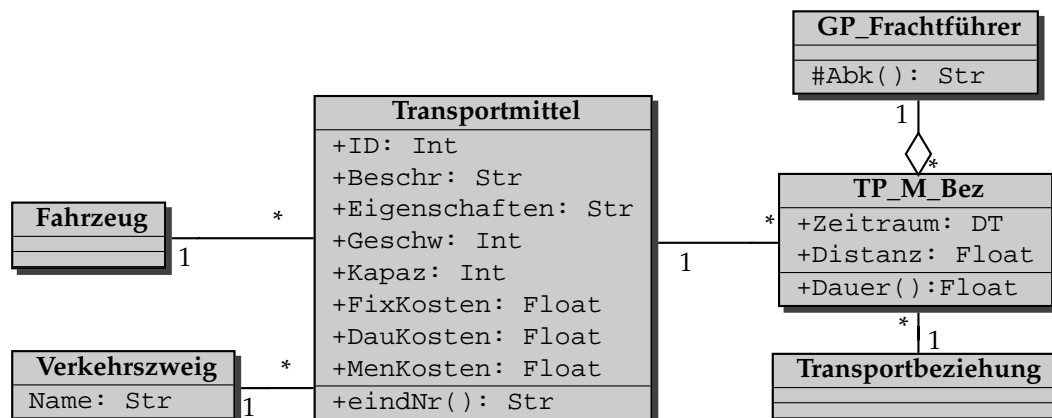
Ersetzt die bereits vorhandene Verbindung zwischen Transportmittel und Transportbeziehungen.

10.5 A27

10.5.1 Anforderung

Ein Transportmittel kann von mehreren Frachtführern [A01] verwendet werden.

10.5.2 UML-Diagramm

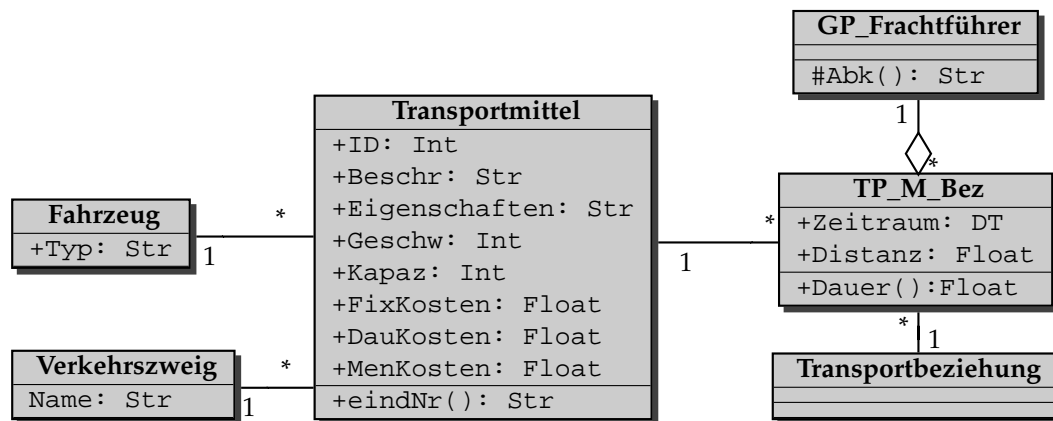


10.6 A28

10.6.1 Anforderung

Fahrzeuge sind bewegliche physikalische Ressourcen (Transportkapazitäten [A49]) wie z. B. Lastwagen, Flugzeuge oder Schiffe, die Waren zwischen Lokationen transportieren können.

10.6.2 UML-Diagramm

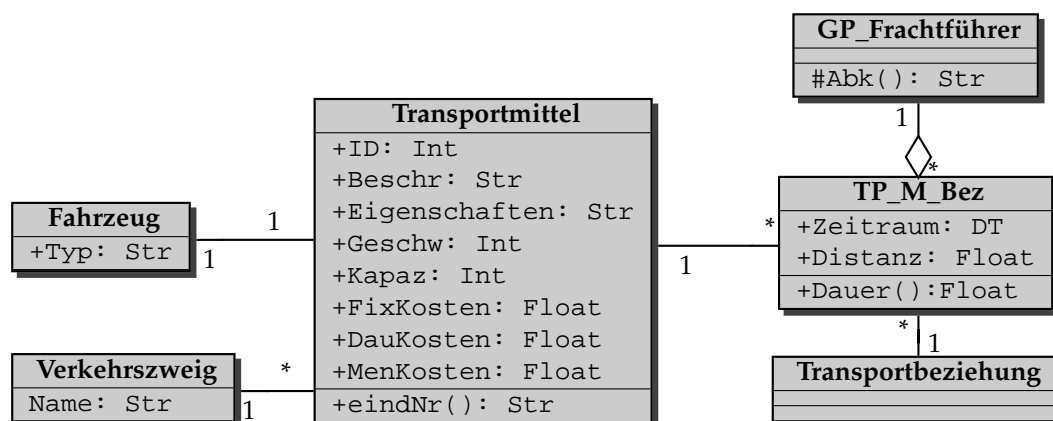


10.7 A29

10.7.1 Anforderung

Jedes Fahrzeug ist durch genau ein Transportmittel klassifiziert.

10.7.2 UML-Diagramm

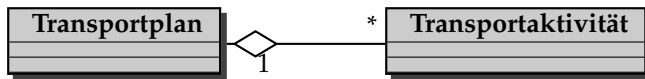


10.8 A30

10.8.1 Anforderung

Das HLS soll die Planung des Transports übernehmen. Ergebnis des Planungsprozesses ist ein Transportplan, welcher Transportaktivitäten enthält.

10.8.2 UML-Diagramm



10.9 A31

10.9.1 Anforderung

Die Erstellung eines Transportplans soll durch das HLS um 2:00 Uhr nachts erfolgen und alle Sendungsanfragen des vorherigen Tages für die Optimierung berücksichtigen.

10.9.2 Anmerkungen

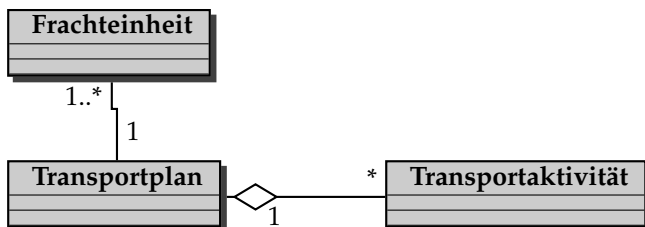
Keine Änderung - Anforderung an Anwendung.

10.10 A32

10.10.1 Anforderung

Ein Transportplan gilt für ein oder mehrere Frachteinheiten [A36].

10.10.2 UML-Diagramm



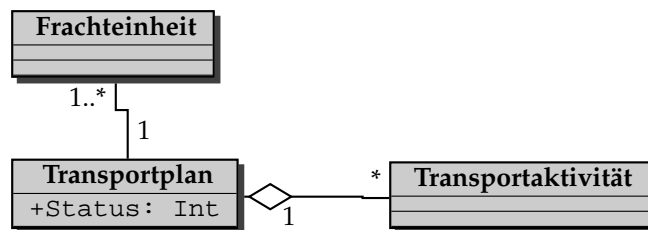
10.11 A33

10.11.1 Anforderung

Ein Transportplan hat einen Status zugeordnet. Mögliche Werte sind:

- „In Planung“: Der Transportplan befindet sich in Planung.
- „Freigegeben“: Der Transportplan ist für die Ausführung freigegeben [A34].
- „Gesperrt“: Die Ausführung des Transportplans ist temporär gesperrt [A35].
- „Beendet“: Der Transportplan wurde abgeschlossen.

10.11.2 UML-Diagramm



10.11.3 Anmerkungen

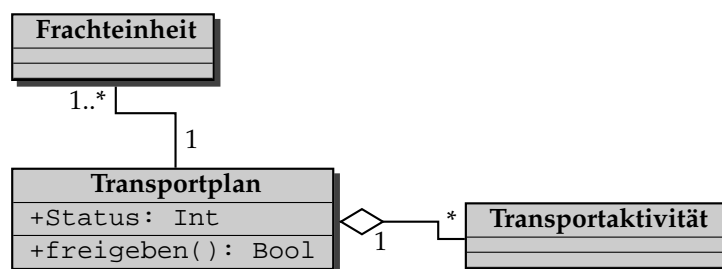
Integer Statuscodes (1-4).

10.12 A34

10.12.1 Anforderung

Ein Transportplan muss durch den Disponenten manuell im System freigegeben werden, sofern der Transportplan noch nicht abgeschlossen wurde.

10.12.2 UML-Diagramm



10.12.3 Anmerkungen

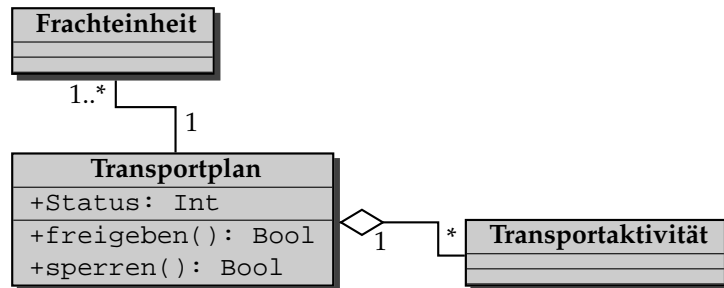
Wahr, wenn erfolgreich freigegeben. Falsch, wenn nicht erfolgreich freigegeben (schon abgeschlossen).

10.13 A35

10.13.1 Anforderung

Der Disponent kann einen Transportplan sperren, sofern der Transportplan noch nicht abgeschlossen wurde. Ein Entsperren geschieht durch eine erneute Freigabe [A32].

10.13.2 UML-Diagramm

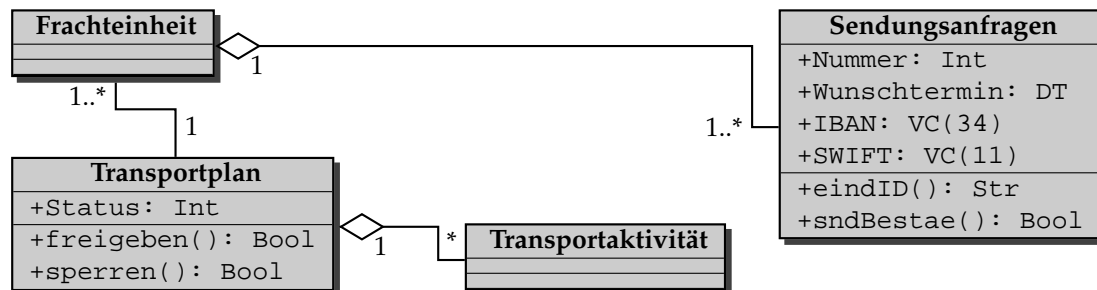


10.14 A36

10.14.1 Anforderung

Eine Frachteinheit stellt einen Transportbedarf dar und ist eine Zusammenstellung von Waren, die gemeinsam durch die gesamte Transportkette transportiert werden. Frachteinheiten werden im Rahmen des Transportplanungsprozesses aus ein oder mehreren Sendungsanfragen gebildet, um die Waren für den Transport geeignet zu bündeln.

10.14.2 UML-Diagramm

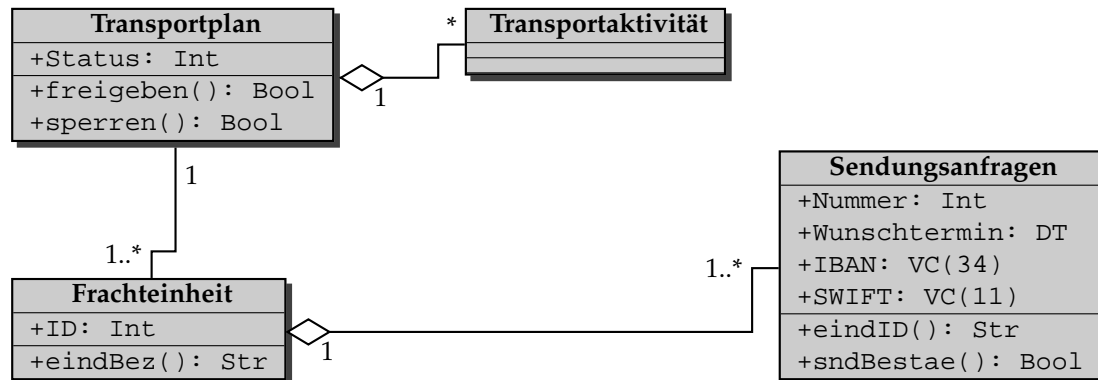


10.15 A37

10.15.1 Anforderung

Eine Frachteinheit ist durch eine eindeutige Bezeichnung identifiziert. Diese hat den Aufbau „FE-<Nummer>“, wobei <Nummer> eine mit dem Wert „1“ beginnende, fortlaufende natürliche Zahl ist. Beispiel: „FE-14“.

10.15.2 UML-Diagramm

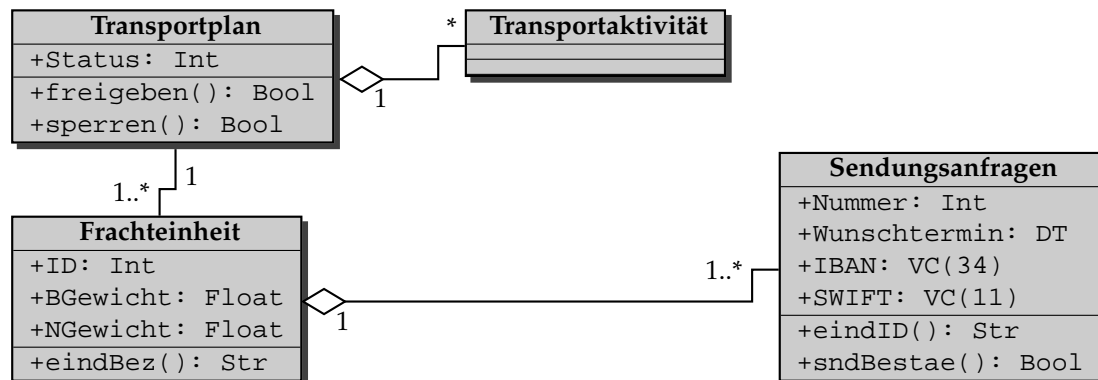


10.16 A38

10.16.1 Anforderung

Eine Frachteinheit hat die Attribute Bruttogewicht und Nettogewicht.

10.16.2 UML-Diagramm

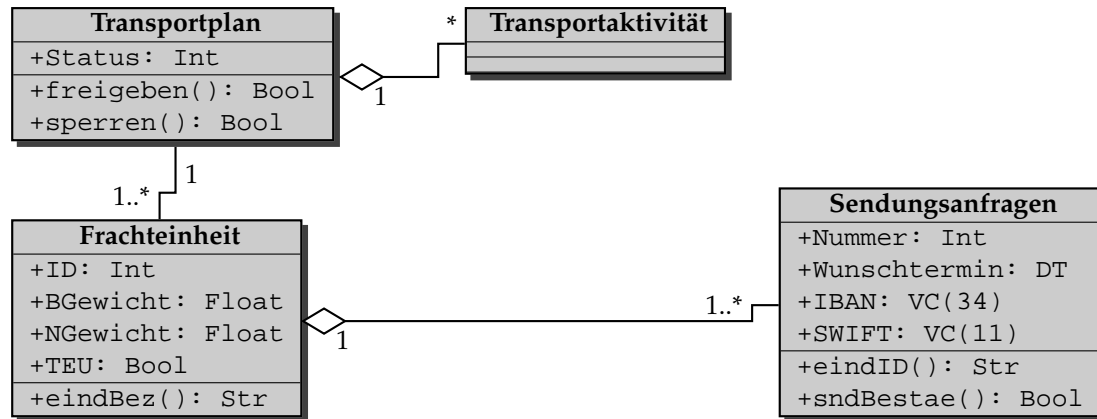


10.17 A39

10.17.1 Anforderung

Eine Frachteinheit ist in HLS stets ein 20- oder ein 40-Fuß Standardcontainer.

10.17.2 UML-Diagramm



10.17.3 Anmerkungen

Wahr = 20-Fuß(TEU). Falsch = 40-Fuß(FEU).

10.18 A40

10.18.1 Anforderung

Ein geeigneter Algorithmus zur Bildung der Frachteinheiten ist vom Auftragnehmer zu konzipieren. Das System soll dabei so wenige Frachteinheiten wie möglich bilden.

10.18.2 Anmerkungen

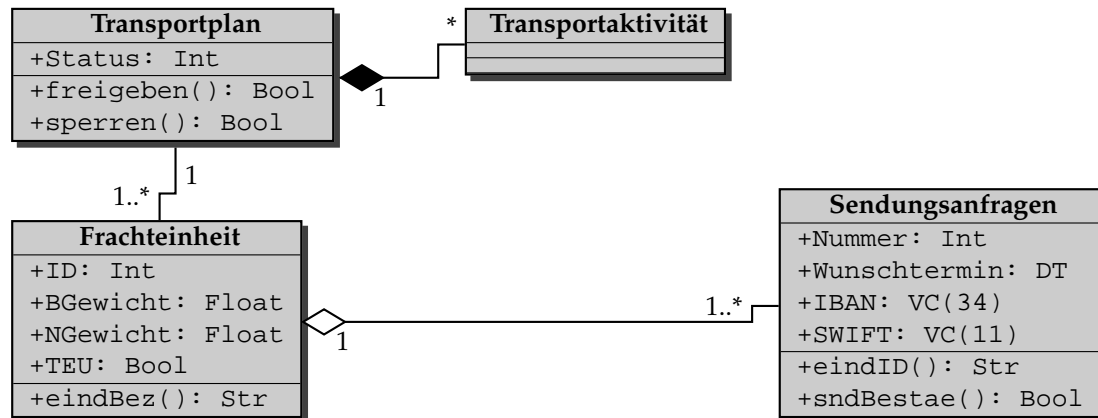
Keine Änderung - Anforderung an Anwendung.

10.19 A41

10.19.1 Anforderung

Ein Transportplan besteht aus Transportaktivitäten. Transportaktivitäten definieren den Einsatz von Transportkapazitäten [2.5.4] ([A48]-[A51]) zur Durchführung der Transportbedarfe [A36].

10.19.2 UML-Diagramm

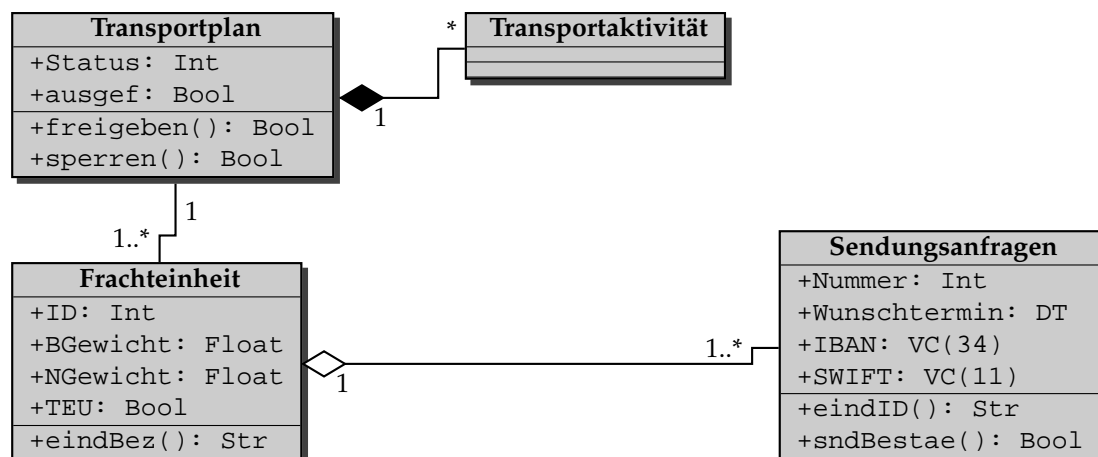


10.20 A42

10.20.1 Anforderung

Es soll im System zwischen geplanten und letztendlich konkret ausgeführten Transportaktivitäten unterschieden werden.

10.20.2 UML-Diagramm



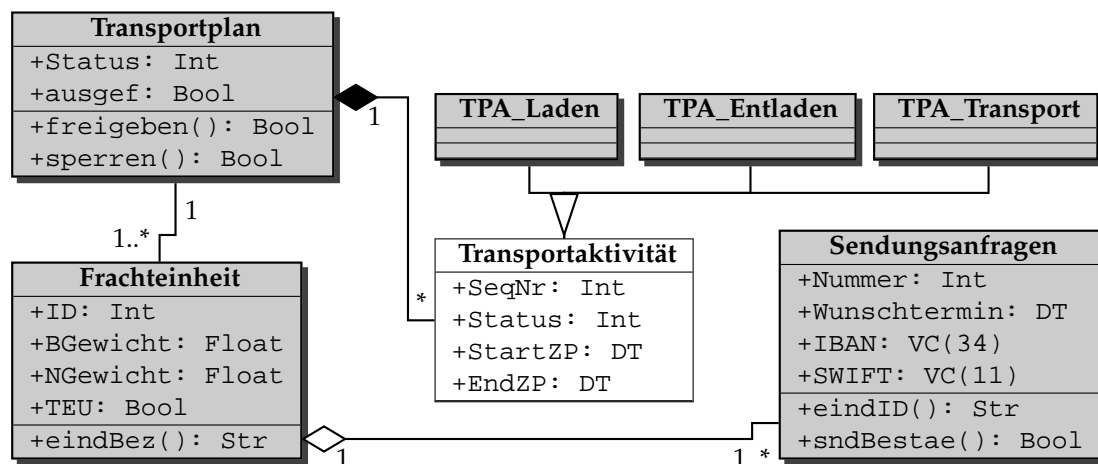
10.21 A43

10.21.1 Anforderung

Eine Transportaktivität besteht aus:

- a. einer Sequenznummer
- b. einer Art:
 - i. „Laden“: Eine Frachteinheit wird auf eine Transportkapazität geladen.
 - ii. „Transport“: Null oder mehrere Frachteinheiten werden durch eine Transportkapazität transportiert.
 - iii. „Entladen“: Eine Frachteinheit wird von einer Transportkapazität entladen.
- c. einem Status:
 - i. „In Planung“: Die Transportaktivität wird geplant.
 - ii. „Freigegeben“: Die Transportaktivität ist für die Ausführung freigegeben.
 - iii. „Gesperrt“: Die Transportaktivität ist von der Ausführung ausgenommen.
 - iv. „Beendet“: Die Transportaktivität wurde abgeschlossen.
- d. einem Start- und einem Endzeitpunkt

10.21.2 UML-Diagramm

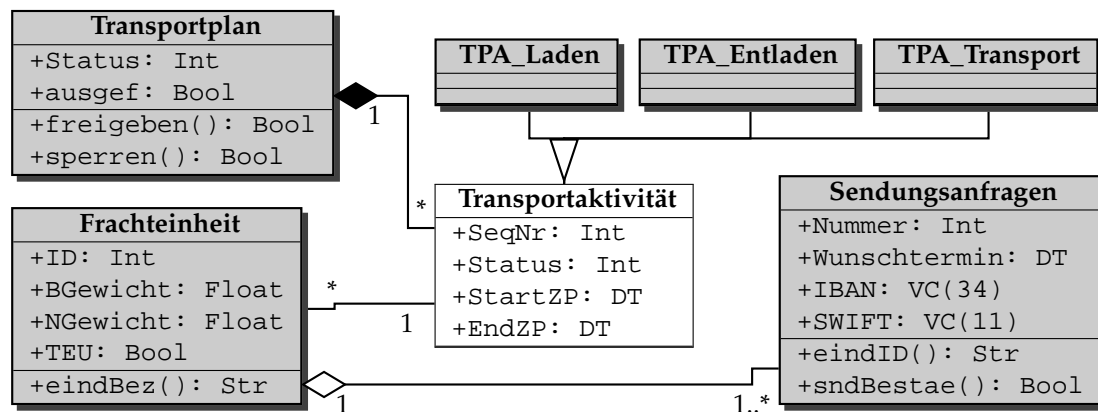


10.22 A44

10.22.1 Anforderung

Eine Transportaktivität kann sich auf mehrere Frachteinheiten beziehen. Beispielsweise kann eine Transportaktivität der Art „Transport“ mehrere Frachteinheiten transportieren.

10.22.2 UML-Diagramm



10.23 A45

10.23.1 Anforderung

Die Sequenznummer gibt eine Reihenfolge der Transportaktivitäten vor. Dies schränkt jedoch nicht die u.U. mögliche parallele Ausführung (z.B. parallele Ladevorgänge unterschiedlicher Frachteinheiten) ein.

10.23.2 Anmerkungen

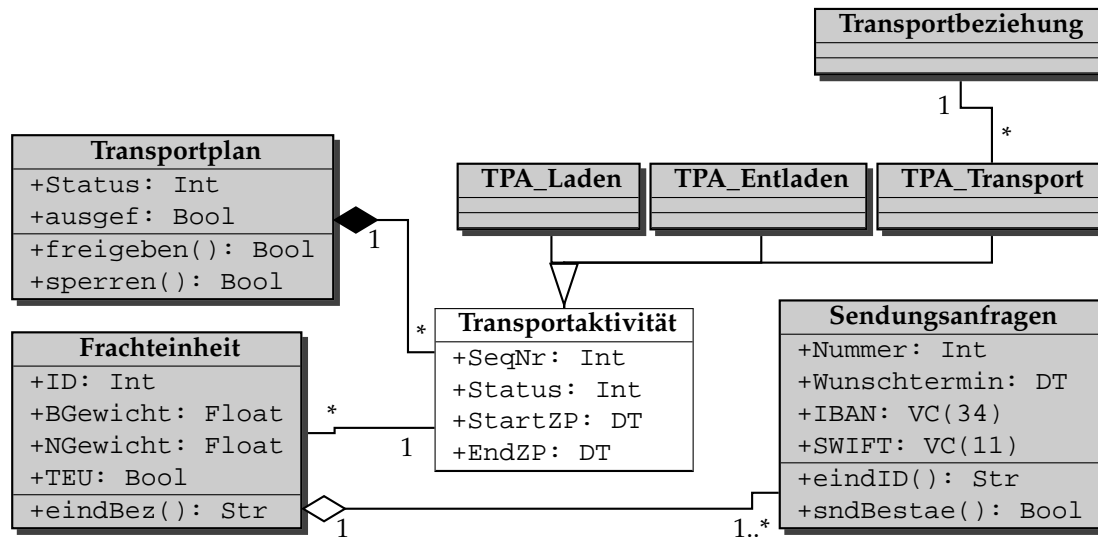
Keine Änderung - Anforderung an Anwendung.

10.24 A46

10.24.1 Anforderung

Eine Transportaktivität der Art „Transport“ ist einer Transportbeziehung zugeordnet, auf der der Transport der Frachteinheiten erfolgen soll bzw. erfolgt ist.

10.24.2 UML-Diagramm

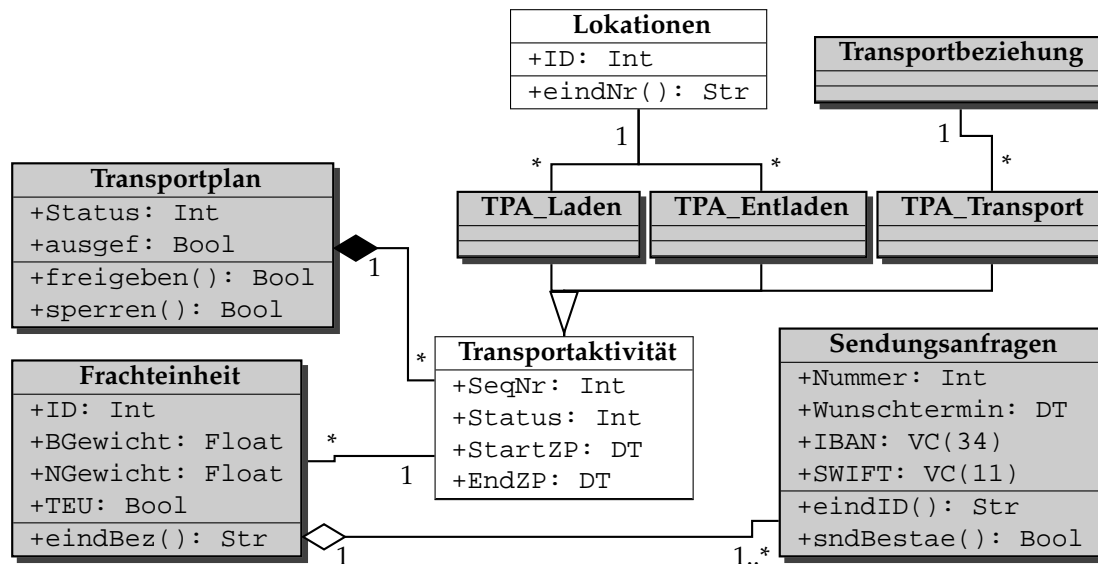


10.25 A47

10.25.1 Anforderung

Eine Transportaktivität der Arten „Laden“ und „Entladen“ ist einer Lokation zugeordnet, an der die Transportaktivität ausgeführt wird.

10.25.2 UML-Diagramm



10.26 A48

10.26.1 Anforderung

Die Planung der Transportaktivitäten eines Transportplans muss die im Folgenden beschriebenen Kapazitätseinschränkungen berücksichtigen.

10.26.2 Anmerkungen

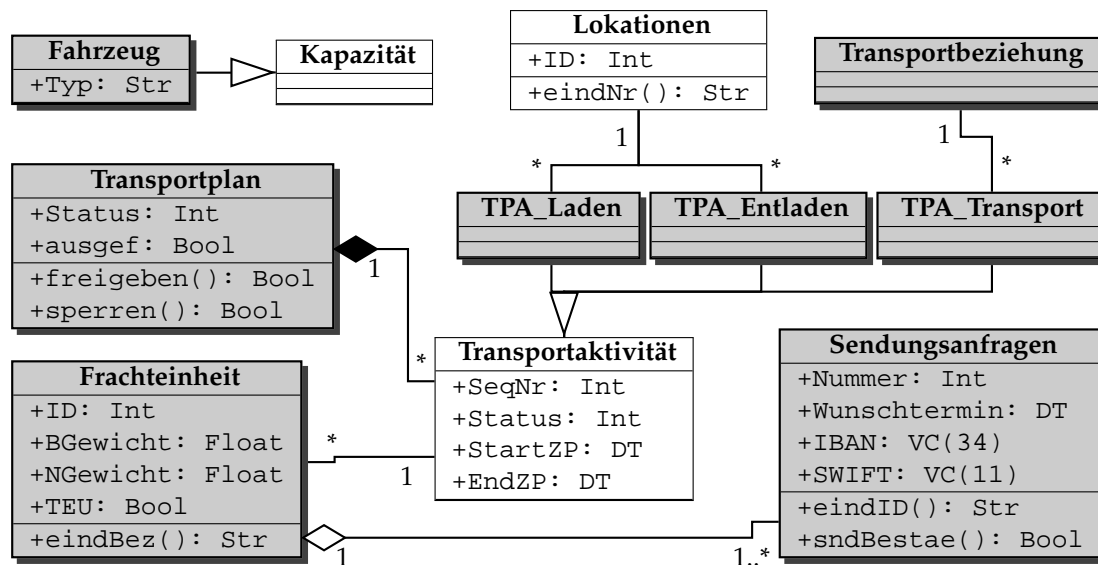
Keine Änderung - Bemerkung zu folgenden Anforderungen.

10.27 A49

10.27.1 Anforderung

Ein Transport kann nur dann stattfinden, wenn ein Fahrzeug mit der entsprechenden Kapazität im Transportzeitraum verfügbar ist. Ein Fahrzeug stellt somit eine Kapazität der Art „Transportkapazität“ dar.

10.27.2 UML-Diagramm

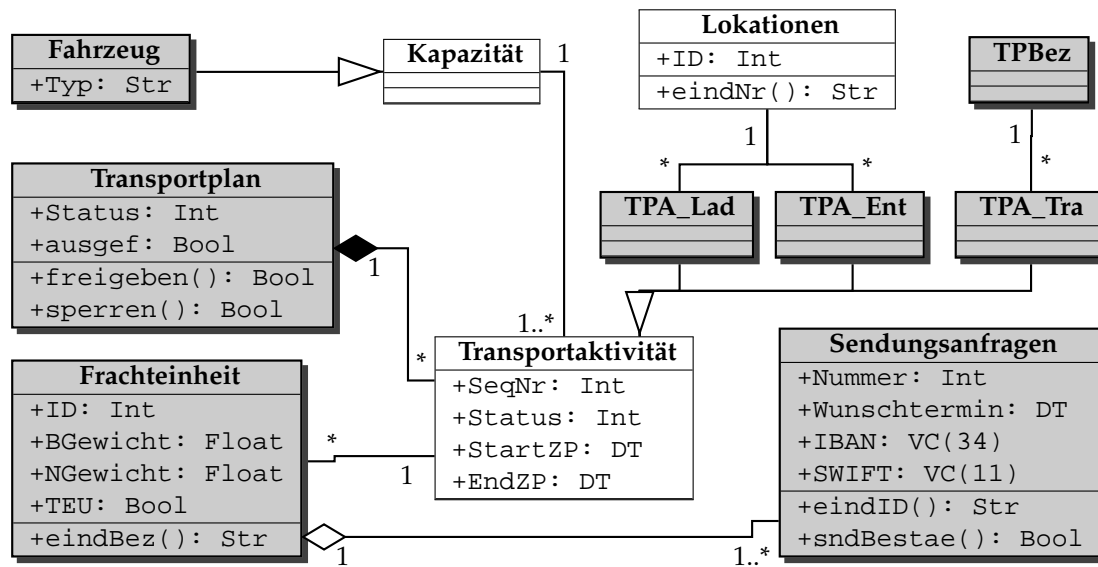


10.28 A50

10.28.1 Anforderung

Einer Transportaktivität sind ein oder mehrere Kapazitäten zugeordnet.

10.28.2 UML-Diagramm



10.28.3 Anmerkungen

Klassennamen etwas abgekürzt.

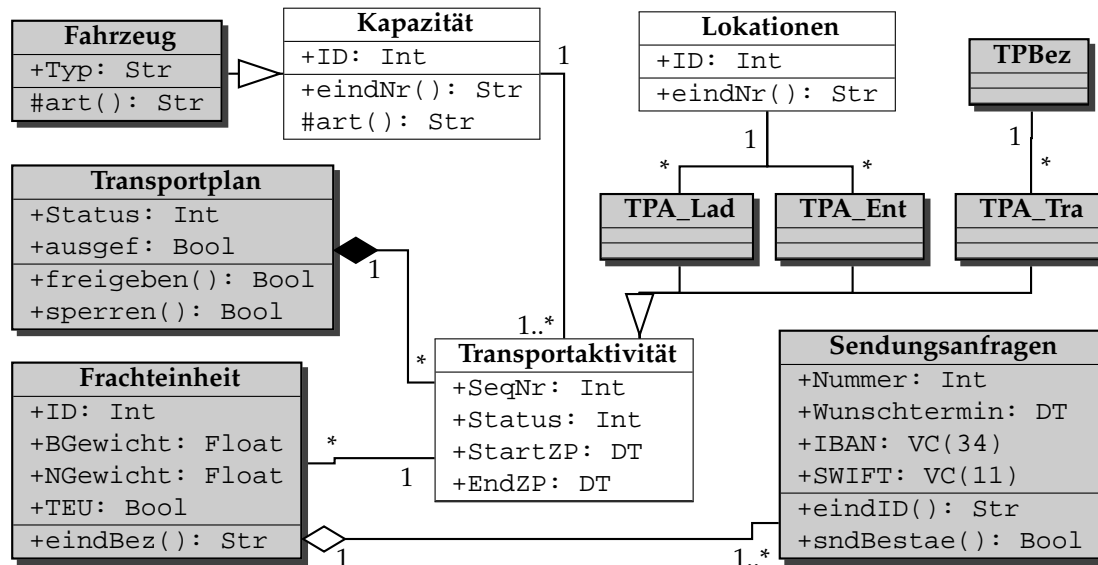
10.29 A51

10.29.1 Anforderung

Jede Kapazität ist durch eine Nummer der Form „KAP-<Art>-n“ (n ist eine natürliche Zahl) eindeutig identifiziert. Die Art ist wie folgt bestimmt:

- Transportkapazität Fahrzeug: „FZG“

10.29.2 UML-Diagramm



10.30 SQL-Skript

```
CREATE TABLE db.GP_Typ (
    GPT_ID INT PRIMARY KEY,
    GPT_Name Varchar(20),
    GPT_Abk Varchar(3)
);

INSERT INTO db.GP_Typ VALUES (1, 'Auftraggeber', 'AG');
INSERT INTO db.GP_Typ VALUES (2, 'Versender', 'VER');
INSERT INTO db.GP_Typ VALUES (3, 'Frachtführer', 'FRA');

CREATE TABLE db.Adresse (
    Adr_ID INT PRIMARY KEY,
    Adr_Strasse Varchar(60) NOT NULL,
    Adr_HausNr Varchar(10) NOT NULL,
    Adr_PLZ Varchar(10) NOT NULL,
    Adr_Land Varchar(50) NOT NULL,
    Adr_Laendercode Char(2) NOT NULL,
    Adr_Position Char(14) NOT NULL
);
```

```

);

CREATE TABLE db.Geschaeftpartner (
    GP_ID INT PRIMARY KEY,
    GPT_ID INT REFERENCES db.gp_typ(GPT_ID),
    Adr_ID INT REFERENCES db.adresse(Adr_ID),
    GP_Name Varchar(30) NOT NULL,
    GP_TelefonNr Varchar(20),
    GP_eMail Varchar(60)
);

CREATE TABLE db.Lok_Art (
    LokA_ID INT PRIMARY KEY,
    Art_Desc Clob
);

INSERT INTO db.Lok_Art VALUES (1,'Empfänger');
INSERT INTO db.Lok_Art VALUES (2,'Umschlag');
INSERT INTO db.Lok_Art VALUES (3,'Versender');

CREATE TABLE db.Lokationen (
    LOK_ID INT PRIMARY KEY,
    Adr_ID INT REFERENCES db.adresse(Adr_ID),
    LokA_ID INT REFERENCES db.Lok_Art(LokA_ID)
);

CREATE TABLE db.Direkte_Zone (
    DKZ_ID INT PRIMARY KEY
);

CREATE TABLE db.PLZ_Zone (
    PLZ_ID INT PRIMARY KEY,
    PLZ INT NOT NULL
);

CREATE TABLE db.Zonen_Typ (
    ZOT_ID INT PRIMARY KEY,
    DKZ_ID INT REFERENCES db.Direkte_Zone(DKZ_ID),
    PLZ_ID INT REFERENCES db.PLZ_Zone(PLZ_ID)
);

CREATE TABLE db.Transportzonen (
    TPZ_ID INT PRIMARY KEY,
    ZOT_ID INT REFERENCES db.Zonen_Typ(ZOT_ID)
);

CREATE TABLE db.Transportabschnitt (
    TPA_ID INT PRIMARY KEY
);

```

```

CREATE TABLE db.LOK_in_TPN (
  LiN_ID INT PRIMARY KEY,
  LOK_ID INT UNIQUE REFERENCES db.Lokationen(LOK_ID)
);

CREATE TABLE db.TPZ_in_TPN (
  ZiN_ID INT PRIMARY KEY,
  TPZ_ID INT UNIQUE REFERENCES db.Transportzonen(TPZ_ID)
);

CREATE TABLE db.Transportnetzwerk (
  TPN_ID INT PRIMARY KEY,
  LiN_ID INT NOT NULL REFERENCES db.LOK_in_TPN(LiN_ID),
  ZiN_ID INT NOT NULL REFERENCES db.TPZ_in_TPN(ZiN_ID)
);

CREATE TABLE db.TPZ_con_LOK (
  CON_ID INT PRIMARY KEY,
  TPZ_ID INT REFERENCES db.Transportzonen(TPZ_ID),
  LOK_ID INT REFERENCES db.Lokationen(LOK_ID)
);

CREATE TABLE db.Quelle (
  Quelle_ID INT PRIMARY KEY,
  LOK_ID INT REFERENCES db.Lokationen(LOK_ID),
  TPZ_ID INT REFERENCES db.Transportzonen(TPZ_ID)
);

CREATE TABLE db.Ziel (
  Ziel_ID INT PRIMARY KEY,
  LOK_ID INT REFERENCES db.Lokationen(LOK_ID),
  TPZ_ID INT REFERENCES db.Transportzonen(TPZ_ID)
);

CREATE TABLE db.Kundenrechnungen (
  KR_ID INT PRIMARY KEY
);

CREATE TABLE db.Kundenfrachtabrechnungen (
  KFR_ID INT PRIMARY KEY,
  KR_ID INT REFERENCES db.Kundenrechnungen(KR_ID)
);

CREATE TABLE db.Waren (
  WAR_ID INT PRIMARY KEY,
  Beschreibung Clob,
  Mengeneinheit Varchar(20),
  Gewicht Float,

```



```

    Volumen Float
);

CREATE TABLE db.Sendungspositionen (
    SP_ID INT PRIMARY KEY,
    WAR_ID INT REFERENCES db.Waren(WAR_ID),
    Menge INT,
    BruttoGewicht Float,
    BruttoVolumen Float
);

CREATE TABLE db.Verkehrszweig (
    VKZW_ID INT PRIMARY KEY,
    VK_Name Varchar(30)
);

INSERT INTO db.Verkehrszweig VALUES (1,'See');
INSERT INTO db.Verkehrszweig VALUES (2,'Luft');
INSERT INTO db.Verkehrszweig VALUES (3,'Land');

CREATE TABLE db.Fahrzeug (
    FZ_ID INT PRIMARY KEY,
    Typ Varchar(20)
);

CREATE TABLE db.Transportmittel (
    TPM_ID INT PRIMARY KEY,
    FZ_ID INT NOT NULL UNIQUE REFERENCES db.Fahrzeug(FZ_ID),
    VKZW_ID INT REFERENCES db.Verkehrszweig(VKZW_ID),
    Beschr Varchar(50),
    Eigenschaften clob,
    Geschw Number(3),
    Kapaz Int,
    FixKosten Float,
    DauKosten Float,
    MenKosten Float
);

CREATE TABLE db.Transportbeziehung (
    TPB_ID INT PRIMARY KEY,
    Ziel_ID INT NOT NULL REFERENCES db.Ziel(Ziel_ID),
    Quell_ID INT NOT NULL REFERENCES db.Quelle(Quelle_ID),
    TPA_Anfang INT REFERENCES db.Transportabschnitt(TPA_ID),
    TPA_Fahrt INT REFERENCES db.Transportabschnitt(TPA_ID),
    TPA_Ende INT REFERENCES db.Transportabschnitt(TPA_ID)
);

CREATE TABLE db.TP_M_Bez (
    TPMB_ID INT PRIMARY KEY,

```

```

    TPM_ID INT REFERENCES db.Transportmittel(TPM_ID),
    TPB_ID INT REFERENCES db.Transportbeziehung(TPB_ID),
    GP_ID INT REFERENCES db.Geschaeftpartner(GP_ID),
    Zeitraum Date,
    Distanz Float
);

CREATE TABLE db.TPA_Transport (
    TPA_Tra_ID INT PRIMARY KEY,
    TPB_ID INT REFERENCES db.Transportbeziehung(TPB_ID)
);

CREATE TABLE db.TPA_Entladen (
    TPA_Ent_ID INT PRIMARY KEY,
    LOK_ID INT REFERENCES db.Lokationen(LOK_ID)
);

CREATE TABLE db.TPA_Laden (
    TPA_Lad_ID INT PRIMARY KEY,
    LOK_ID INT REFERENCES db.Lokationen(LOK_ID)
);

CREATE TABLE db.Kapazitaet (
    KAP_ID INT PRIMARY KEY,
    FZ_ID INT REFERENCES db.Fahrzeug(FZ_ID),
    Art Varchar(20)
);

CREATE TABLE db.Transportplan (
    TRP_ID INT PRIMARY KEY,
    Status INT,
    Ausgef Number(1) NOT NULL
);

CREATE TABLE db.Transportaktivitaet (
    TPA_ID INT PRIMARY KEY,
    SeqNr INT,
    Status INT,
    StartZP Date,
    EndZP Date,
    TPA_Tra_ID INT REFERENCES db.TPA_Transport(TPA_Tra_ID),
    TPA_Ent_ID INT REFERENCES db.TPA_Entladen(TPA_Ent_ID),
    TPA_Lad_ID INT REFERENCES db.TPA_Laden(TPA_Lad_ID),
    KAP_ID INT REFERENCES db.Kapazitaet(KAP_ID),
    TRP_ID INT REFERENCES db.Transportplan(TRP_ID) NOT NULL
);

CREATE TABLE db.Frachteinheit (
    FRE_ID INT PRIMARY KEY,

```

```

    TPA_ID INT REFERENCES db.Transportaktivitaet(TPA_ID),
    BGewicht Float,
    NGewicht Float,
    TEU Number(1) NOT NULL
);

CREATE TABLE db.Sendungsanfragen (
    SA_ID INT PRIMARY KEY,
    Wunschtermin Date,
    IBAN Varchar(34),
    SWIFT Varchar(11),
    GP_ID INT NOT NULL REFERENCES db.Geschaeftpartner(GP_ID),
    Abgangsort_LOK_ID INT REFERENCES db.Lokationen(LOK_ID),
    Zielort_LOK_ID INT NOT NULL REFERENCES db.Lokationen(LOK_ID),
    KFR_ID INT NOT NULL UNIQUE
    REFERENCES db.Kundenfrachtabrechnungen(KFR_ID),
    SP_ID INT NOT NULL REFERENCES db.Sendungspositionen(SP_ID),
    FRE_ID INT REFERENCES db.Frachteinheit(FRE_ID)
);

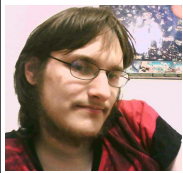
```

11 Aufgabe 11 (SQL)

11.0 Aufgabenstellung

Experimentieren Sie mit den bisher erlernten SQL-Befehlen auf Ihrer Instanz der DB herum. Hier ist keine Dokumentation erforderlich.

Informationen zur Signatur

	Unterzeichner	EMAILADDRESS=robin.ladiges@haw-hamburg.de, CN=Robin Christopher Ladiges
	Datum/Zeit	Sun Jun 27 00:12:48 CEST 2010
	Austeller-Zertifikat	CN=CAcert Class 3 Root, OU=http://www.CAcert.org, O=CAcert Inc.
	Serien-Nr.	44727
	Methode	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signatur)