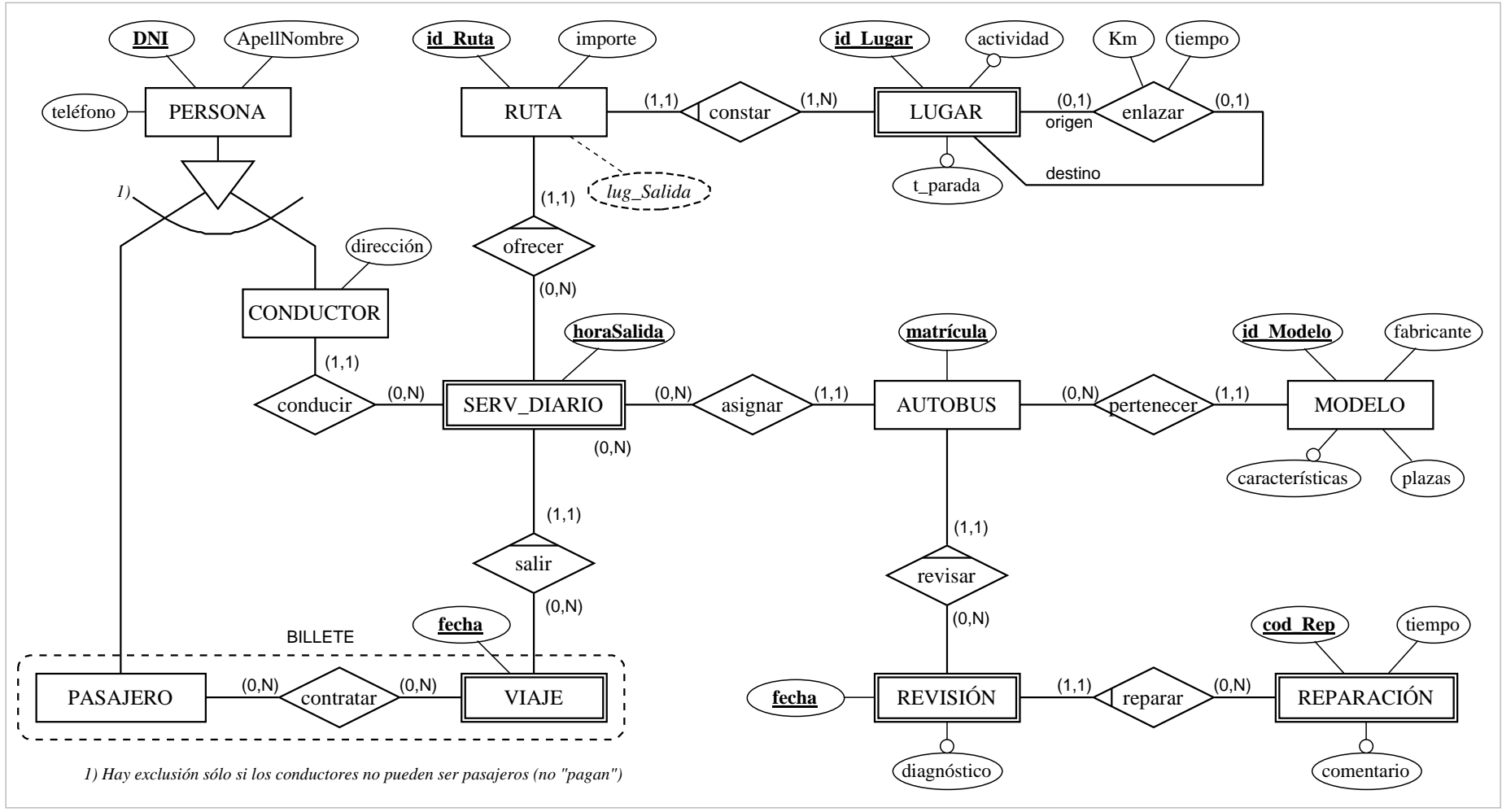


Esquema Entidad-Interrelación (1)

Dominios: . . . (véase esquema relacional)



Restricciones:

- ☛ verificar que todo lugar de una ruta enlaza con otro de la misma ruta, o con ninguno (es el último).
- ☛ verificar que todas los lugares de una ruta forman una lista cuyo primer lugar es el lug_Salida (ee el único que no tiene lugar anterior).
- ☛ verificar que el total de pasajeros de un viaje es \leq n° de plazas disponibles en el autobús asignado
- ☛ etc.

Esquema relacional de la Base de Datos TuryBus

DOMINIOS

tpNombre = cadena[32];
tpTexto = cadena[255];
tpActividad = { comida, descanso, visita };

ESQUEMAS DE RELACION

Ruta (Id_Ruta : tpNombre;
lug_Salida : tpNombre, **NO NULO**;
importe : entero, **NO NULO**;
(Id_Ruta, lug_Salida), **clave ajena de** Lugar);

Lugar (id_Ruta : tpNombre, **clave ajena de** Ruta;
id_Lugar : tpNombre;
sig_Lugar : tpNombre;
actividad : tpActividad;
t_parada : entero;
Km : entero;
tiempo : entero;
(id_Ruta, sig_Lugar), **UNICO**, **clave ajena de** Lugar);

- ☛ verificar que si sigLugar es NULO, km y tiempo son NULOS; en caso contrario, son NO NULOS.
- ☛ verificar que todo lugar de la ruta es accesible desde el lugar origen de la ruta.

Modelo (Id_Modelo : tpNombre;
fabricante : tpNombre, **NO NULO**;
numPlazas : entero, **NO NULO**;
características : tpNombre);

Autobús (matricula : tpNombre;
Id_Modelo : tpNombre, **NO NULO**, **clave ajena de** Modelo);

Revisión (fecha : tpFecha;
matricula : tpNombre, **NO NULO**, **clave ajena de** Autobús;
diagnóstico : tpTexto);

Reparación (fecha : tpFecha;
matricula : tpNombre, **clave ajena de** Autobús;
codReparacion : tpNombre;
tiempo : entero, **NO NULO**;
comentario : tpTexto;
(fecha, matricula), **clave ajena de** Revisión);

Conductor (DNI : tpNombre;
ApellNombre : tpNombre, **NO NULO**;
teléfono : entero, **NO NULO**;
dirección : tpNombre, **NO NULO**);

ServDiario (id_Ruta : tpNombre, **clave ajena de** Ruta;
horaSalida : entero;
matricula : tpNombre, **NO NULO**, **clave ajena de** Autobús;
chófer : tpNombre, **NO NULO**, **clave ajena de** Conductor);

Viaje (id_Ruta : tpNombre;
horaSalida : entero;
fecha : tpFecha;
(id_Ruta, horaSalida), **clave ajena de** ServDiario);

Pasajero (DNI : tpNombre;
ApellNombre : tpNombre, **NO NULO**;
teléfono : entero, **NO NULO**);

Contratar (id_Ruta : tpNombre;
horaSalida : entero;
fecha : tpFecha;
DNI : tpNombre, **clave ajena de** Pasajero;
(id_Ruta, horaSalida, fecha), **clave ajena de** Viaje);

- ☛ verificar que el total de pasajeros de un viaje es \leq n° de plazas disponibles en el autobús asignado

☛ verificar que: $\prod_{DNI}(\text{Conductor}) \cap \prod_{DNI}(\text{Pasajero}) = \emptyset$