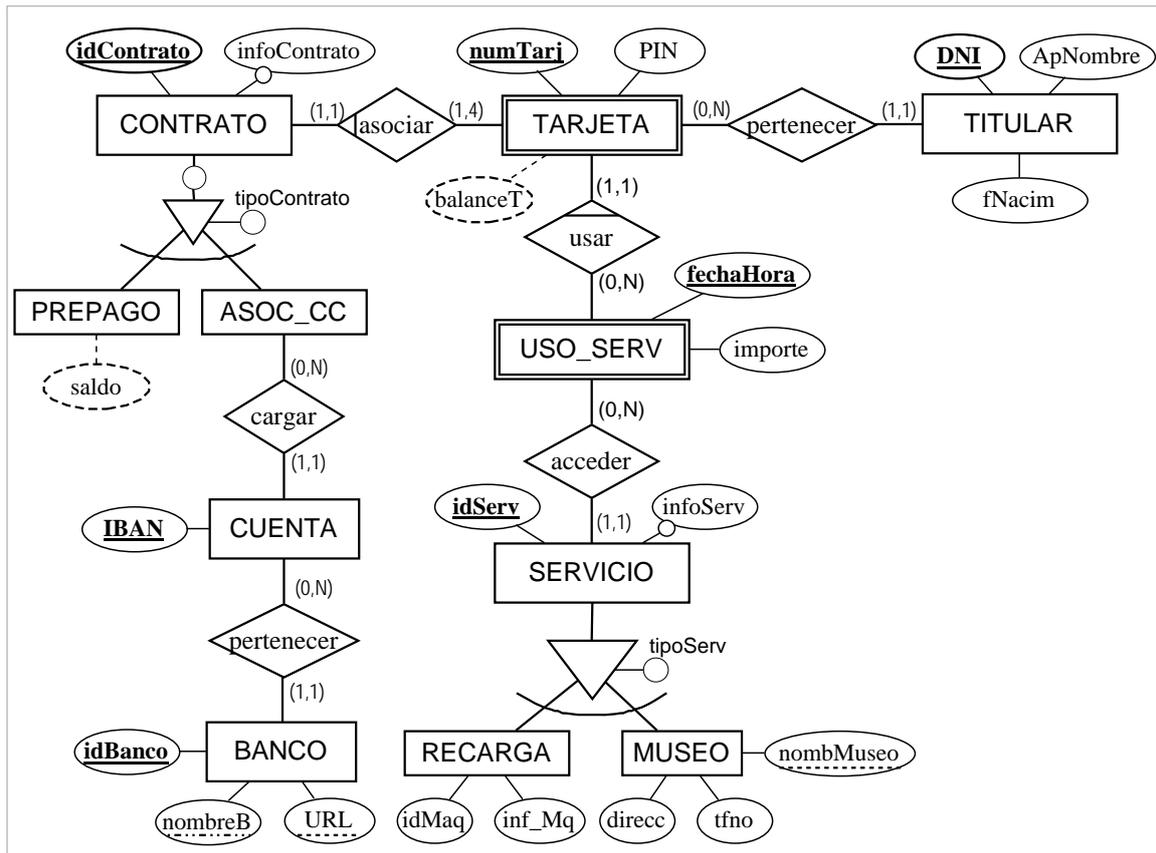


Esquema E/R de la base de datos "tarjetas ciudadanas"

DOMINIOS: (véase esquema relacional) ••••

••••

RESTRICCIONES: •••• (sólo alguna a modo de ejemplo)

- * **verificar que** \forall contrato \exists tarjeta asociada con número 1 {la del titular del contrato}
- * **verificar que** \forall tarjeta se verifica que $1 \leq \text{NumTarj} \leq 4$.
- * **verificar que** \forall tarjeta $[\text{idContrato}, \text{numTarj}] \mid \text{numTarj} > 1 \exists$ la tarjeta $[\text{idContrato}, \text{numTarj}-1]$.
- * **verificar que** \forall tarjeta $[\text{idContrato}, 1]$ la edad del titular asociado es ≥ 18 años.
- * **verificar que** \forall titular se verifica que $1 \leq \text{edad} \leq 120$.
- * **verificar que** \forall contrato prepago, $\text{saldo} = \sum$ importes de los servicios usados por las tarjetas asociadas al contrato
- * **verificar que** \forall contrato prepago, $\text{saldo} \geq 0$ {no se permitirán accesos a servicios que generen un saldo negativo}
- * **verificar que** \forall tarjeta asociada a contrato prepago, $\text{balanceT} = \sum$ importes de los servicios usados por esa tarjeta
- * **verificar que** \nexists uso de servicio de recarga para tarjetas asociadas a cuenta bancaria.
- *

••••

Esquema relacional de la base de datos "tarjetas ciudadanas"**DOMINIOS**

```
tpNombre = cadena(24);          tpInfo = cadena(64);          tpSeq = 1..9999; {autoincrementado}
tpContrato = (prepago, asoc_CC);  tpServicio = ('recarga', 'autobus', 'museo');
```

ESQUEMAS DE RELACION**BANCO (**

```
idBanco : tpNombre;
nombreB : tpNombre, UNICO, NO NULO;
URL : tpNombre, UNICO, NO NULO
```

);

CUENTA (

```
IBAN : tpNombre;
idBanco : tpNombre, NO NULO, referencia a BANCO(idBanco), borrado en CASCADA
```

);

CONTRATO (

```
idContrato : natural;
tipoContrato : tpContrato, NO NULO;
saldo : natural;
IBAN : tpNombre, referencia a CUENTA(IBAN)
```

```
* verificar que Si tipoContrato = prepago
    entonces saldo es NO NULO e IBAN es NULO
    si no    saldo es NULO e IBAN es NO NULO
```

Fsi;

```
* verificar que  $\forall$ ocurrencia de CONTRATO,  $\exists$  TARJETA[idContrato,1]
```

);

TITULAR (

```
DNI : tpNombre;
ApNombre : tpNombre, NO NULO;
fNacim : tpFecha, NO NULO, check(fNacim < HOY and HOY - fNacim < 120)
```

);

TARJETA (

```
idContrato : natural, referencia a CONTRATO(idContrato), borrado en CASCADA;
numTarj : natural, check(numTarj > 0 and numTarj <=4);
PIN : natural, NO NULO;
DNI_titular : tpNombre, NO NULO, referencia a TITULAR(DNI)
balanceT : natural; { balanceT es NO NULO sii CONTRATO[idContrato].tipoContrato=prepago }
(idContrato, DNI_titular) UNICO
```

);

SERVICIO (

```
idServ : natural;
tipoServ : tpServicio, NO NULO;
infoServ : tpInfo
```

);

MUSEO (

```
idServ : natural, referencia a SERVICIO(idServ), borrado en CASCADA;
nombMuseo : tpNombre, UNICO, NO NULO;
direccion : tpNombre, NO NULO;
tfno : natural, NO NULO
```

```
* verificar que  $\forall$ ocurrencia de MUSEO, SERVICIO[idServ].tipoServ = 'museo'
```

);

ACCESOS (

```
idContrato : natural;
numTarj : natural;
fechaHora : tpFecha;
idServ : natural, NO NULO, referencia a SERVICIO(idServ), borrado en CASCADA;
importe : real, NO NULO;
(idContrato, numTarj) clave ajena de TARJETA
```

);

RESTRICCIONES ••••###

```
* verificar que ... {resto de restricciones del esquema E/R: edad titular contrato > 18 año, etc.}
```