

UML

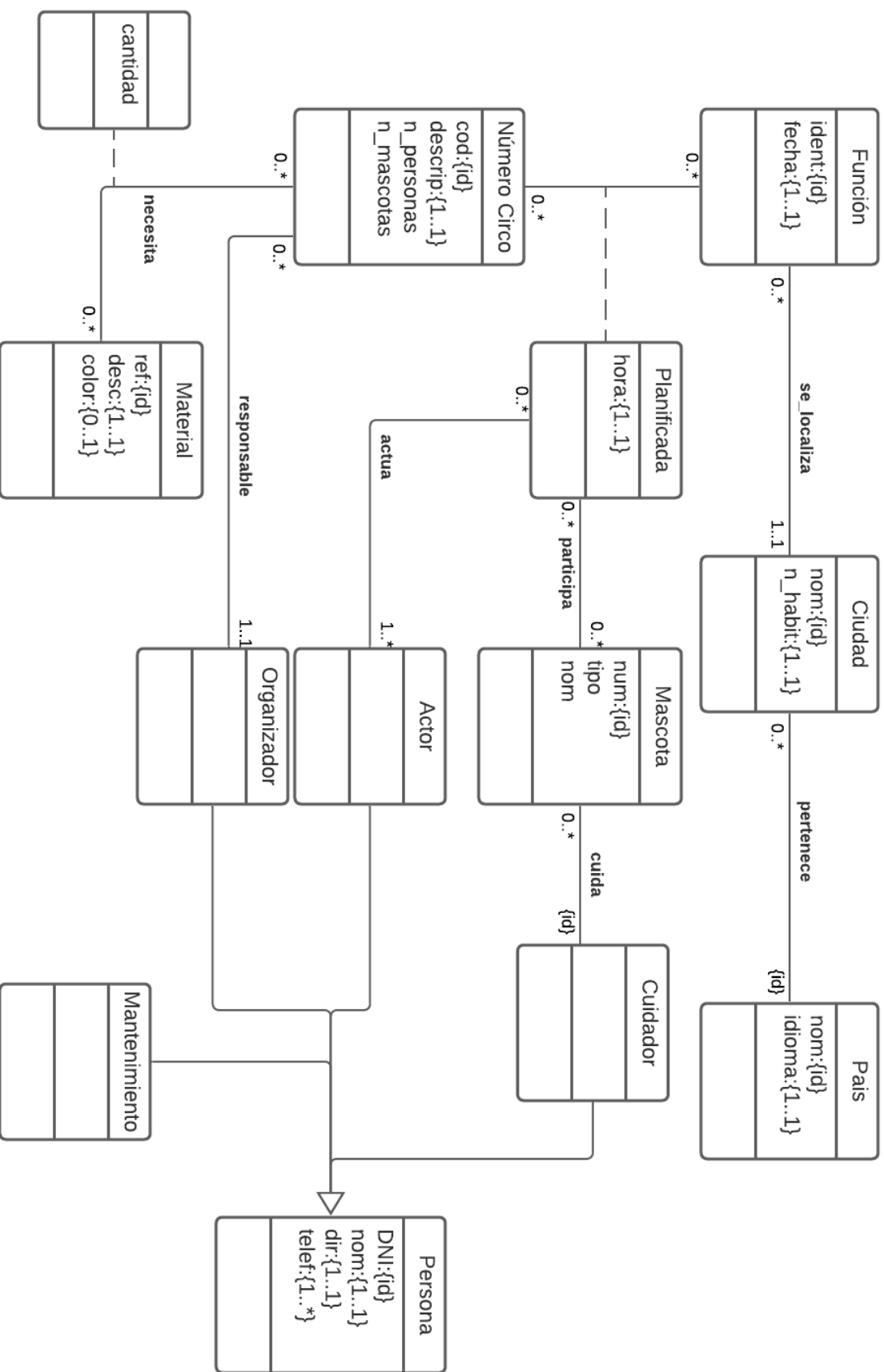
Una compañía circense desea crear una base de datos para planificar la programación de sus funciones. Para ello dispone de un catálogo de los diferentes números del circo que sabe realizar dicha compañía. De cada número se conoce el código que lo identifica, la descripción, la cantidad de actores y de animales necesarios, la persona responsable de organizar el número y, opcionalmente, la relación de los materiales y la cantidad que se necesitará de cada uno de éstos. Los materiales se identifican con una referencia, y se tiene una descripción y el color de éste.

A cada función diaria planificada se le asigna un número que la identifica, la fecha y la ciudad donde se celebrará. Cada ciudad se identifica por un nombre, e interesa saber el número de habitantes. Puede que dos ciudades tengan el mismo nombre, aunque en este caso corresponderán a distintos países. De los países se debe conocer tanto el nombre, que los identifica, como el idioma que mayoritariamente se habla.

Según las circunstancias y la disponibilidad del personal, para cada función se planifican los diferentes números de los que se compondrá, y para cada uno de ellos se registrará la hora programada y los actores que participarán, al menos uno.

También se detallarán las mascotas que participarán, en el caso en que el número lo requiera. De cada mascota se guarda el nombre, el tipo y el número de identificación que corresponderá a un número correlativo en función del cuidador que tenga asignado.

De cada una de las personas que trabajan en el circo se registra el DNI que lo identifica, el nombre, la dirección, los teléfonos y las funciones que puede desempeñar. Las funciones que puede realizar el personal son actuar, organizar, cuidar mascotas o mantener las instalaciones.



Normalización

Sea el siguiente esquema de relación:

R (A: entero, B: texto, C: entero, D: conjunto enteros, E: texto, F: texto, G: texto, H: texto)

CP: {A, B, C}

VNN: {D, E, F, G, H}

Teniendo en cuenta las dependencias que se exponen a continuación, transfórmela a un conjunto de relaciones en tercera forma normal.

$\{B, C\} \rightarrow \{F\}$ $\{F\} \rightarrow \{G\}$ $\{A\} \rightarrow \{E\}$

SOLUCION Normalización.

R1(A: entero, B: texto, C: entero, H:texto) CP:{A,B,C} VNN:{H} CA:{B,C}-> R3 CA:{A}->R5	R2(A: entero, B: texto, C: entero, D:entero) CP:{A,B,C,D} CA: {A,B,C}->R1	R3(B: texto, C: entero, F: texto) CP:{B,C} CA:{F}->R4 VNN:{F}
RI: Todo valor de A,B,C debe estar en R2	R4 (F: texto, G: texto) CP:{F} VNN:{G}	R5 (A: entero, E: texto) CP:{A} VNN:{E}

Sea el siguiente esquema de relación:

R (X: entero, Y: texto, Z: entero, V: conjunto enteros, W: texto, Q: texto, P: texto, R: texto)

CP: {X, Y, Z}

VNN: {V, W, Q, P, R}

Teniendo en cuenta las dependencias que se exponen a continuación, transfórmela a un conjunto de relaciones en tercera forma normal.

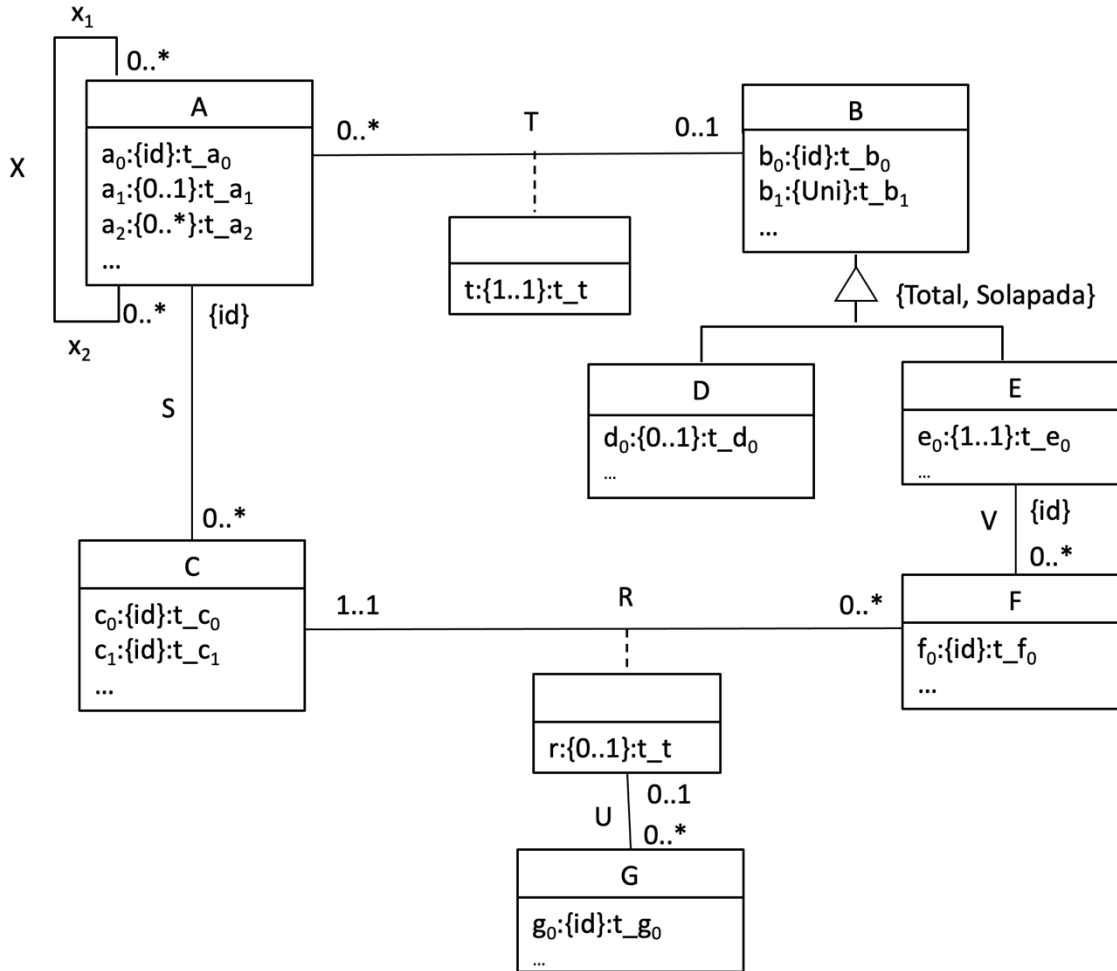
$\{Y, Z\} \rightarrow \{Q\}$ $\{Q\} \rightarrow \{P\}$ $\{X\} \rightarrow \{W\}$

SOLUCION Normalización.

R1(X: entero, Y: texto, Z: entero, R:texto) CP:{X,Y,Z} VNN:{R} CA:{Y,Z}-> R3 CA:{X}->R5	R2(X: entero, Y: texto, Z: entero, V:entero) CP:{X,Y,Z,V} CA: {X,Y,Z}->R1	R3(Y: texto, Z: entero, Q: texto) CP:{Y,Z} CA:{Q}->R4 VNN:{Q}
RI: Todo valor de X,Y,Z debe estar en R2	R4 (Q: texto, P: texto) CP:{Q} VNN:{P}	R5 (X: entero, W: texto) CP:{X} VNN:{W}

Diseño lógico

Realice el diseño lógico del siguiente diagrama de clases en UML para obtener un conjunto equivalente de relaciones del modelo relacional. No es necesario indicar los tipos de datos en las relaciones. Las restricciones que no pueda expresar en el esquema relacional, escríbalas en lenguaje natural



A ($a_0:t_{a_0}, a_1:t_{a_1}, \dots$)
CP:{ a_0 }

A1 ($a_0:t_{a_0}, a_2:t_{a_1}$)
CP:{ a_0, a_2 }
CAj:{ a_0 } \rightarrow A

C ($c_0:t_{c_0}, c_1:t_{c_1}, a_0:t_{a_0}, \dots$)
CP:{ c_0, c_1, a_0 }
CAj:{ a_0 } \rightarrow A

B ($b_0:t_{b_0}, b_1:t_{b_1}, \dots$)
CP:{ b_0 }
UNI:{ b_1 }

D ($b_0:t_{b_0}, d_0:t_{d_0}, \dots$)
CP:{ b_0 }
CAj:{ b_0 } \rightarrow B

E ($b_0:t_{b_0}, e_0:t_{e_0}, \dots$)
CP:{ b_0 }
CAj:{ b_0 } \rightarrow B
VNN:{ e_0 }

R.I._{total}: Todo valor de b_0 de B debe estar en b_0 de D o en b_0 de E

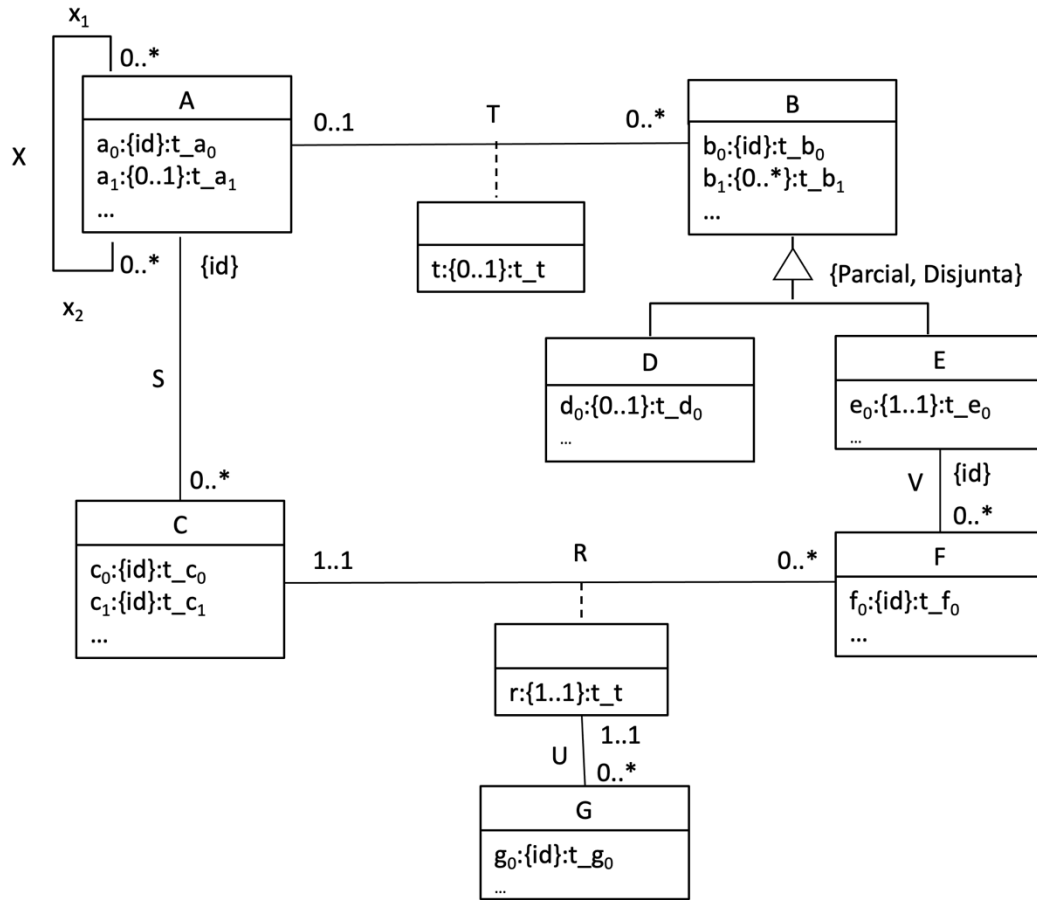
F ($f_0:t_{f_0}, b_0:t_{b_0}, c_0:t_{c_0}, c_1:t_{c_1}, a_0:t_{a_0}, r:t_r, \dots$)
CP:{ f_0, b_0 }
CAj:{ b_0 } \rightarrow E
CAj:{ c_0, c_1, a_0 } \rightarrow C
VNN:{ c_0, c_1, a_0 }

G ($g_0:t_{c_0}, f_0:t_{f_0}, b_0:t_{b_0}, \dots$)
CP:{ g_0 }

CAj:{ f_0, b_0 } \rightarrow F(f_0, b_0)

X ($a_{0 \rightarrow x1}:t_{a_0}, a_{0 \rightarrow x2}:t_{a_0}$)
CP:{ $a_{0 \rightarrow x1}, a_{0 \rightarrow x2}$ }
CAj:{ $a_{0 \rightarrow x1}$ } \rightarrow A
CAj:{ $a_{0 \rightarrow x2}$ } \rightarrow A

T ($a_0:t_{a_0}, b_0:t_{b_0}, t:t_t$)
CP:{ a_0 }
CAj:{ a_0 } \rightarrow A
CAj:{ b_0 } \rightarrow B
VNN:{ b_0, t }



A ($a_0:t_{a_0}, a_1:t_{a_1}, \dots$)
CP:{ a_0 }

C ($c_0:t_{c_0}, c_1:t_{c_1}, a_0:t_{a_0}, \dots$)
CP:{ c_0, c_1, a_0 }
CAj:{ a_0 } \rightarrow A

B ($b_0:t_{b_0}, \dots$)
CP:{ b_0 }

B1 ($b_0:t_{b_0}, b_1:t_{b_1}$)
CP:{ b_0, b_1 }
CAj:{ b_0 } \rightarrow B

D ($b_0:t_{b_0}, d_0:t_{d_0}, \dots$)
CP:{ b_0 }
CAj:{ b_0 } \rightarrow B

E ($b_0:t_{b_0}, e_0:t_{e_0}, \dots$)
CP:{ b_0 }
CAj:{ b_0 } \rightarrow B
VNN:{ e_0 }

R.I._{disjunta}: No puede haber ningún valor de b_0 de B que esté en b_0 de D y en b_0 de E

F ($f_0:t_{f_0}, b_0:t_{b_0}, c_0:t_{c_0}, c_1:t_{c_1}, a_0:t_{a_0}, r:t_r, \dots$)
CP:{ f_0, b_0 }
CAj:{ b_0 } \rightarrow E
CAj:{ c_0, c_1, a_0 } \rightarrow C
VNN:{ c_0, c_1, a_0, r }

G ($g_0:t_{c_0}, f_0:t_{f_0}, b_0:t_{b_0}, \dots$)
CP:{ g_0 }
CAj:{ f_0, b_0 } \rightarrow F
NNV:{ f_0, b_0 }

X ($a_{0-x1}:t_{a_0}, a_{0-x2}:t_{a_0}$)
CP:{ a_{0-x1}, a_{0-x2} }
CAj:{ a_{0-x1} } \rightarrow A
CAj:{ a_{0-x2} } \rightarrow A

T ($a_0:t_{a_0}, b_0:t_{b_0}, t:t_t$)
CP:{ b_0 }
CAj:{ b_0 } \rightarrow B
CAj:{ a_0 } \rightarrow A
VNN:{ a_0 }