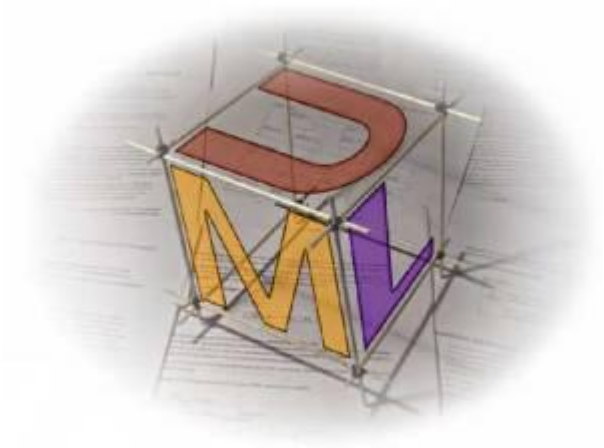
Enunciado

Crear un proyecto UML llamado **Asociacion** en el que se diseñe un diagrama de clases que modele el proceso de **dar de alta a cada una de las personas que se apuntan a una asociación**.



De cada **persona** interesa saber sus **datos básicos**: **NIF**, **nombre completo** y **fecha de nacimiento**. Cuando cada nuevo **socio** se da de alta, se le asigna un **código de asociado** alfanumérico y se anota la **fecha de alta**.

La clase **Fecha** se modela con tres campos (**día**, **mes** y **año**) de tipo entero. La clase **Nif** se modela con un campo de tipo entero llamado **dni** y un campo de tipo carácter llamado **letra**.

Análisis del enunciado

El primer paso a realizar consiste en leer detenidamente el enunciado y extraer de el toda la información posible. A veces es cuestión de aplicar el sentido común, a veces es cuestión de unir piezas, a veces es cuestión de lógica y a veces es cuestión de pura deducción, pero **siempre** es cuestión de razonar por aproximaciones sucesivas y de experiencia.

Bien, parece que el enunciado refiere únicamente un modelado de datos, no de comportamiento, por lo que se procederá a realizar una lista de los elementos más significativos para el proyecto que se puedan extraer del enunciado.

1. Nombre del proyecto – **Asociacion**
2. Nombre del diagrama – **AltaAsociacion**
3. Ítems – Elementos significativos del enunciado.

- Persona
- Socio
- Nif
- Nombre completo
- Fecha de nacimiento
- Código de asociado
- Día
- Mes
- Año
- Dni
- Letra

4. Tipos de datos
   - Integer
   - Char
   - String
   - Nif
   - Fecha
   - Nombre

Diseño de clases

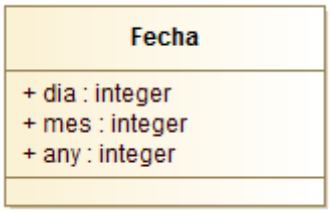
Recuérdese que las clases son entidades que encapsulan información, se trata por tanto de ver qué información de la lista anterior está relacionada entre sí y ver la forma de encapsularla en sus respectivas clases.

Se procederá a identificar las clases a partir del enunciado y de encapsular en ellas la información relacionada. Este paso se realizará considerando las clases de forma aislada las unas de las otras. Posteriormente, cuando se vean las relaciones, se depurará su composición.

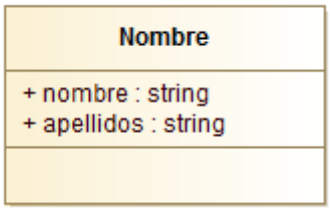
En esta fase del modelado se procede siempre desde las clases más triviales a las más complejas.

Clase Nif

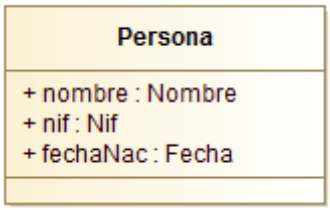
Clase Fecha



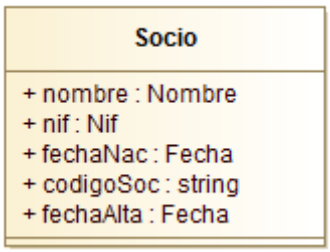
Clase Nombre



Clase Persona



Clase Socio



Relaciones

En esta fase se va a evaluar qué clases tienen que ver con qué otras, es decir sus relaciones. Para que el procedimiento resulte lo más sencillo posible se estudiarán las **relaciones dos a dos**.

Herencia

Primero se abordan las **relaciones de herencia** empezando por aquellas que resulten triviales o más evidentes.

Aunque estrictamente hablando no es así del todo, la **regla para detectarlas** es ver si entre las clases definidas en el diseño existe alguna cuyos atributos sean un **subconjunto** de alguna otra.

Persona – Socio

En este caso resulta que los atributos de la clase **Persona** son un **subconjunto** de los de la clase **Socio** y **semánticamente** tiene sentido que la clase **Socio** sea una **especialización** de la clase **Persona**.

Obsérvese que los atributos que hereda la clase especializada no se representan. Obsérvese también que la flecha que representa esta relación va desde la clase hija a la clase madre, tiene linea continua punta de flecha cerrada, no tiene cardinalidad y no está etiquetada por ningún rol.

Asociación

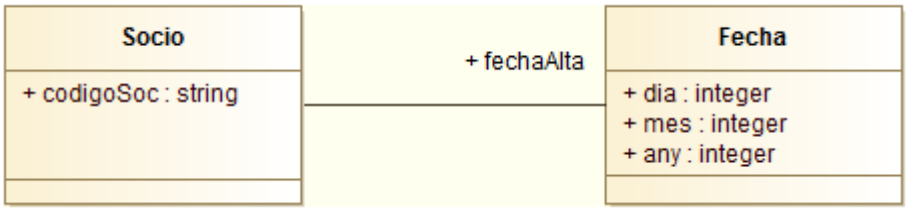
Una vez se han resuelto las relaciones de herencia le toca el turno a los demás tipos de relaciones que son asociaciones. Se procederá siempre abordando primero las triviales o más simples y continuando por las demás. Para que resulte más claro, el análisis se realizará considerando las clases dos a dos.

Socio – Fecha

Aun a riesgo de resultar tedioso pero con el objetivo de que resulte lo más clarificador posible, el análisis de la relación entre estas dos clases se realizará paso a paso.

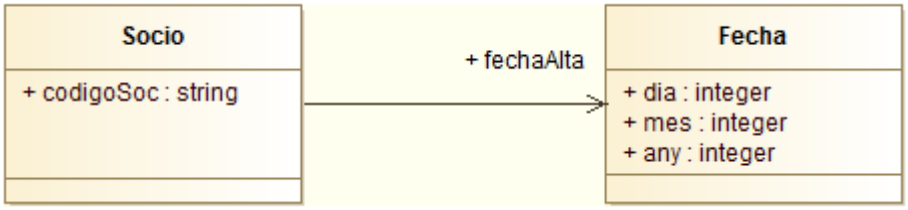
Roles

Esta asociación es evidente. La clase **Socio** tiene un campo de tipo **Fecha**, dicho de otra manera, la clase **Socio** tiene una **referencia** a un objeto de la clase **Fecha**. Así considerado este campo pasa a ser el **rol de la relación** que vincula a ambas clases. Por lo tanto, **desaparece de la clase Socio y aparece en la linea de vinculación junto a la clase de su tipo**.



Navegabilidad

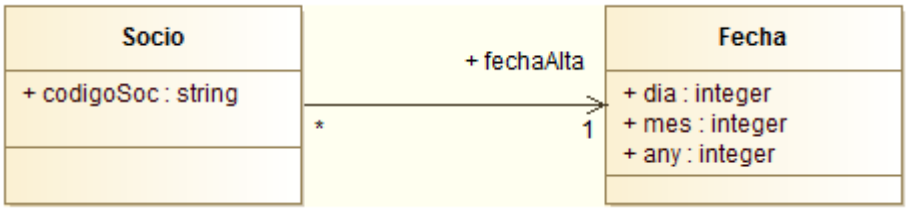
Ahora hay que abordar la **navegabilidad** tratando de ver si desde una clase se puede ir a la otra. Es evidente que la clase **Fecha** no tiene información de la clase **Socio** por lo que la navegabilidad desde la clase **Fecha** no es posible. Sin embargo, la clase **Socio** tiene una referencia a la clase **Fecha** por lo que si es viable la navegabilidad en este sentido. **La navegabilidad se expresa con una punta de flecha abierta puesta en el lado de la clase a la que se llega**.



Cardinalidades

El siguiente paso es abordar las **cardinalidades o multiplicidades**, es decir el **número de instancias de cada clase que intervienen en la relación**. Para resolver este paso hay que preguntar: "¿Por **cada instancia** de una de las dos clases **cuantas instancias** de la otra clase pueden en extremo intervenir **como mínimo** (**Cardinalidad mínima**) y **como máximo** (**Cardinalidad máxima**)?". Y luego **hacer las preguntas al revés**.

- Cuántas fechas de alta como mínimo tiene cada socio : 1
- Cuántas fechas de alta como máximo tiene cada socio: 1
- Cuántos socios se dan de alta como mínimo en una fecha: 0
- Cuántos socios se dan de alta como máximo en una fecha: Varios



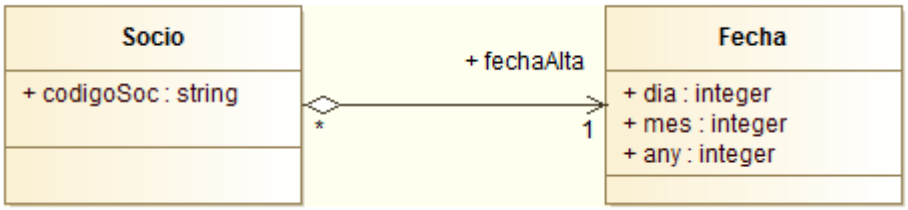
Obsérvese que cuando la **cardinalidad mínima y máxima coinciden** sólo se representa una de ellas. Obsérvese también que cuando la **cardinalidad máxima** es **múltiple** y la **cardinalidad mínima** es **cero** refiere una **cardinalidad múltiple opcional** y se representa con un **asterisco**.

Todo – Parte

El siguiente paso consiste en considerar qué clase es **PARTE** y qué clase es **TODO**. Dicho de otro modo **quien contiene a quien**. En este caso la discriminación es trivial: la clase **Socio** es la parte **TODO** porque tiene una referencia a la clase **Fecha** que es la parte **PARTE**.

Agregación – Composición

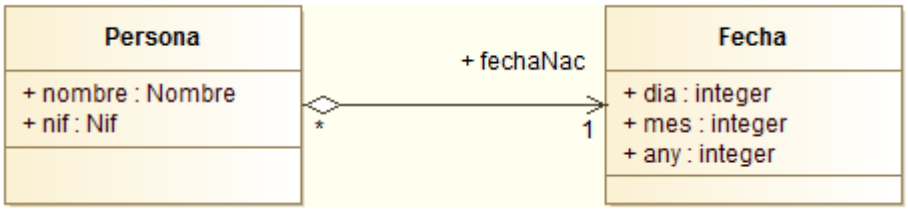
El siguiente paso consiste en determinar si la relación entre las clases es de **agregación** o de **composición**. Para que la relación sea de composición es **condición necesaria** que la **cardinalidad de la parte TODO sea 1**. Como este no es el caso la relación es de agregación.



Obsérvese que el rombo se ha representado en blanco.

Persona – Fecha

El mismo razonamiento empleado para relacionar las clases **Socio** y **Fecha** se puede emplear para relacionar las clases **Persona** y **Fecha**.

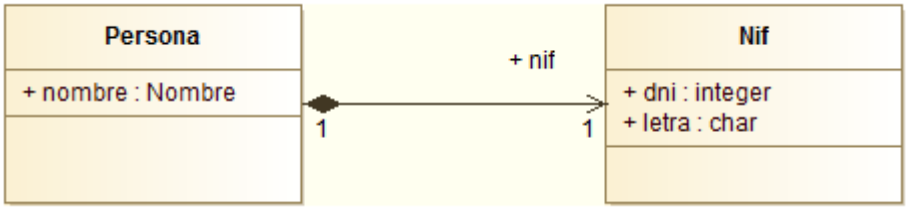


Esta vez el **rol** de la clase **Fecha** en la relación cambia. Obsérvese como ha desaparecido el campo correspondiente a la **fecha de nacimiento** de la clase **Persona**.

Persona – Nif

El análisis de la relación entre estas dos clases determina que cada objeto de la clase **Nif** está unívocamente unido a un solo objeto de la clase **Persona**, y viceversa, por lo que la cardinalidad en ambos lados es la unidad. tanto mínima como máxima.

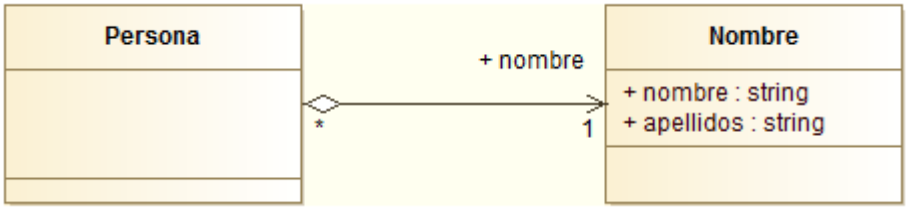
Además **semánticamente** si desaparece la parte **TODO**, el objeto de la clase **Persona**, la existencia de la parte **PARTE** ya no tiene sentido y debería desaparecer también. Esta **dependencia existencial** apunta a una **relación de tipo Composición**.



Obsérvese que el rombo se ha representado relleno en negro. Obsérvese también que el campo correspondiente al Nif ha desaparecido de la clase persona pasando a ser el rol de la relación.

Persona – Nombre

La relación entre la clase **Persona** y la clase **Nombre** es muy parecida a la relación existente entre la clase **Persona** y la clase **Fecha**.



Obsérvese que al trasladar el campo **nombre** al rol de la relación, el diagrama que representa la clase **Persona** ya no contiene ningún atributo.

Diagrama de clases completo

Bueno, ahora se trata de ponerlo **todo junto en un solo diagrama**.