

# Sistemas de representación estructurados

Redes Semánticas y Frames

# Sistemas de representación estructurados

- 1.- Introducción
- 2.- Redes Semánticas
  - 2.1.- Herencia en Redes Semánticas
  - 2.2.- Excepciones en la Herencia
- 3.- Redes Semánticas Extendidas
  - Ejemplos Redes Semánticas
- 4.- Frames
  - 4.1.- Jerarquía o taxonomía de frames
  - 4.2.- Definición de frame.
  - 4.3.- Equivalencia frames / redes semánticas
  - 4.4.- Herencia simple
  - 4.5.- Extensión de la Definición de Frame
  - 4.6.- Herencia múltiple
  - Ejemplo Frames

# Introducción

- Representación del conocimiento mediante grafos (conceptos, relaciones).
- Facilitan la representación del conocimiento humano
  - Redes semánticas
  - Frames

# Redes Semánticas

- Redes Semánticas (R. Quillian, 1968)
  - representación en procesamiento de lenguaje natural
  - formalismo muy limitado para dominios más complejos
  - limitado para tratar con formas de inferencia sofisticada
  - precursor de las frames

# Redes Semánticas

◆ Una red semántica se representa como un grafo dirigido etiquetado (en algunos casos se exige que dicho grafo sea aciclico), constituido por:

- nodos: representan conceptos (un objeto individual o una clase de objetos)
- arcos: representan relaciones binarias entre los conceptos.

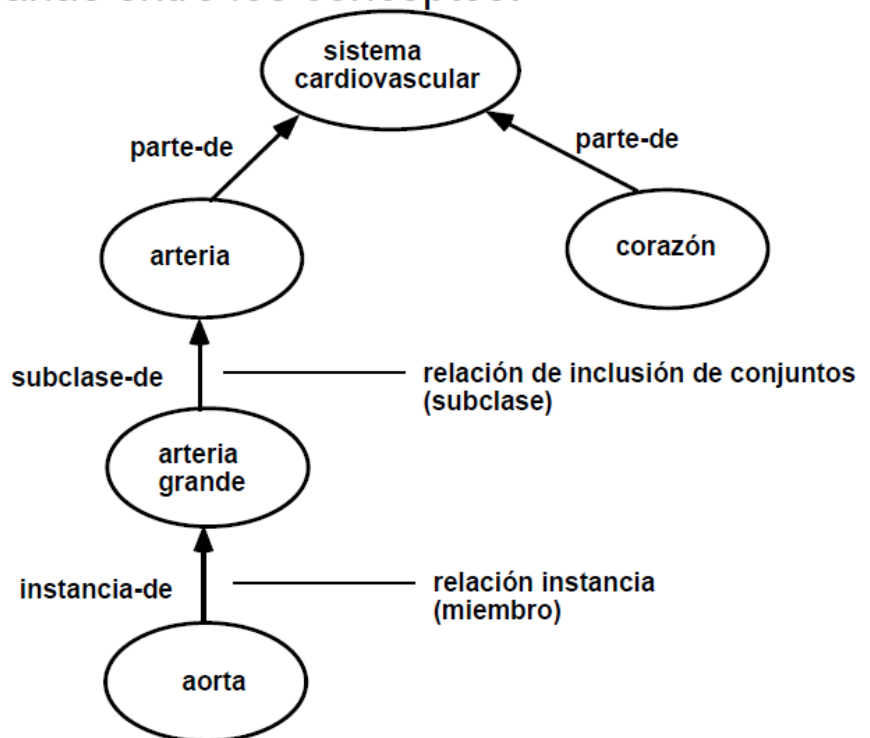
◆ Ejemplo:

"El corazón es parte del sistema cardiovascular"

"Las arterias son parte del sistema cardiovascular"

"Las arterias grandes son arterias"

"La aorta es una arteria"

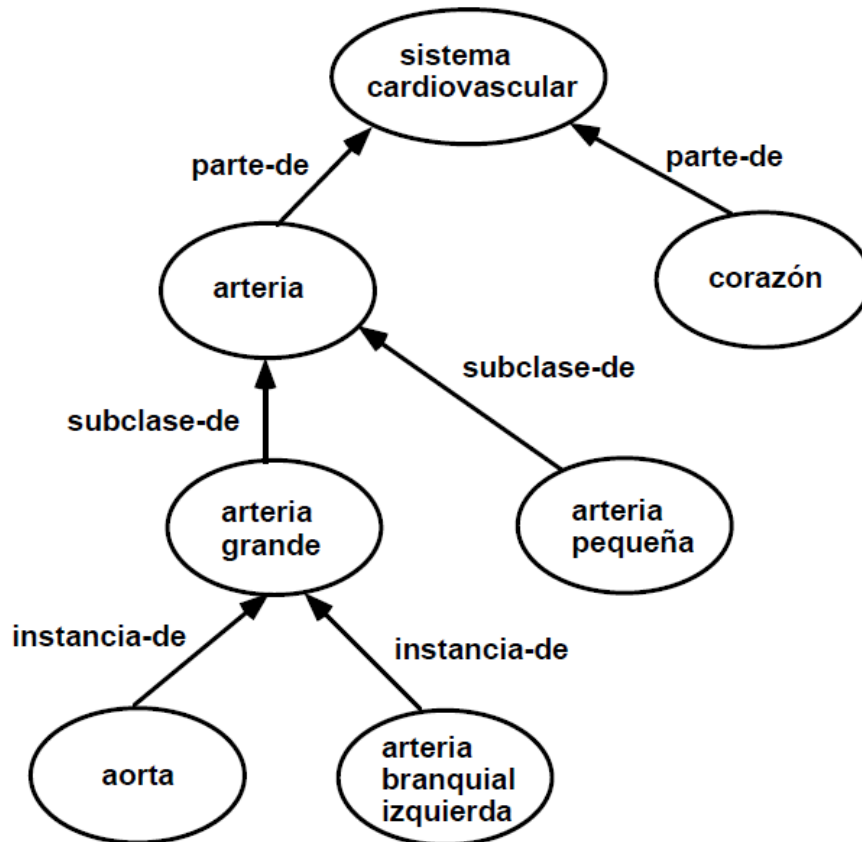


# Redes Semánticas

◆ Ejemplo:

"Las arterias pequeñas son arterias"

"La arteria branquial izquierda es una arteria grande"



- ambigüedad de la relación es-un para distinguir entre las relaciones subclase e instancia (miembro).

- relación subclase (subclase-de)

- relación instancia (instancia-de)

# Herencia en redes semánticas

◆ La Herencia es el mecanismo de razonamiento utilizado en redes semánticas

◆ Herencia: un concepto (nodo) hereda las propiedades de los conceptos "más altos en la jerarquía" a través de las relaciones subclase-de e instancia-de.

◆ Ejemplo:

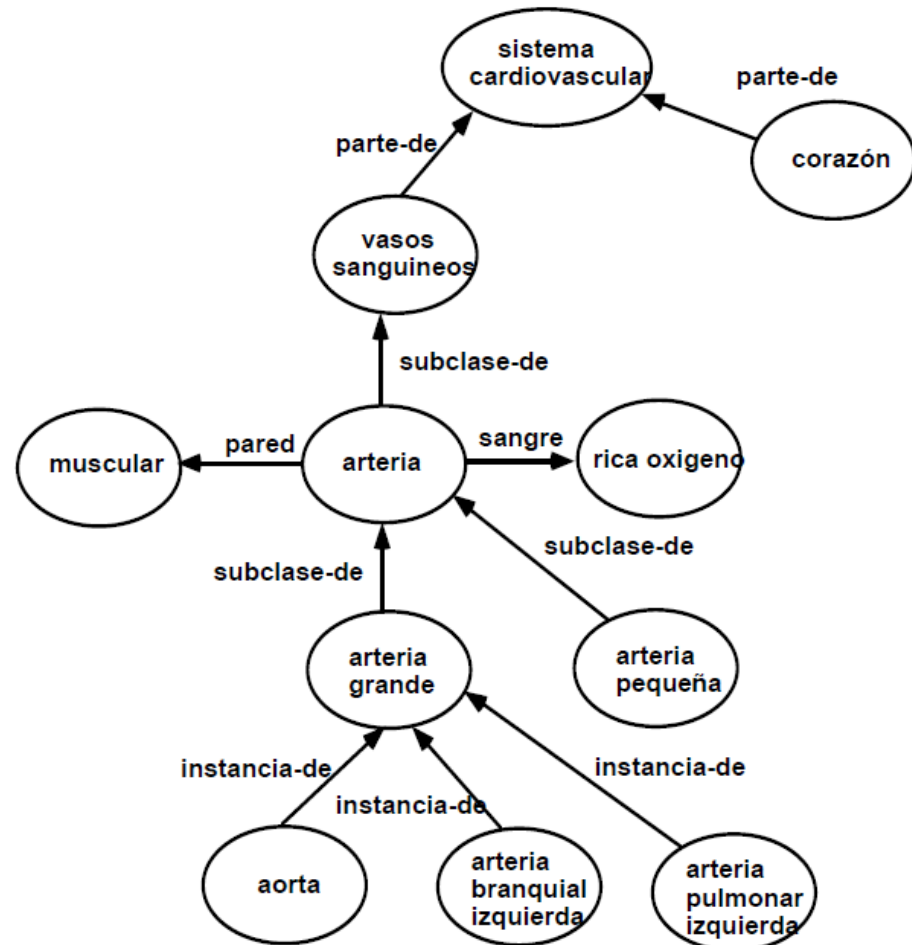
"Un vaso sanguíneo es parte del sistema cardiovascular"

"Las arterias son vasos sanguíneos"

"Las arterias contienen sangre rica en oxígeno"

"Las arterias tienen pared muscular"

"La arteria pulmonar izquierda es una arteria grande"



◆ A partir de la red semántica podemos deducir:

"Las arterias grandes son ricas en oxígeno" / "Las arterias grandes tienen pared muscular" /  
"La aorta contiene sangre rica en oxígeno" / "La aorta tiene pared muscular"

# Excepciones en la herencia

a) No heredar propiedades que producen inconsistencias.

"La arteria pulmonar izquierda contiene sangre pobre en oxígeno"

"La arteria pulmonar izquierda tiene pared muscular y es rica en oxígeno"

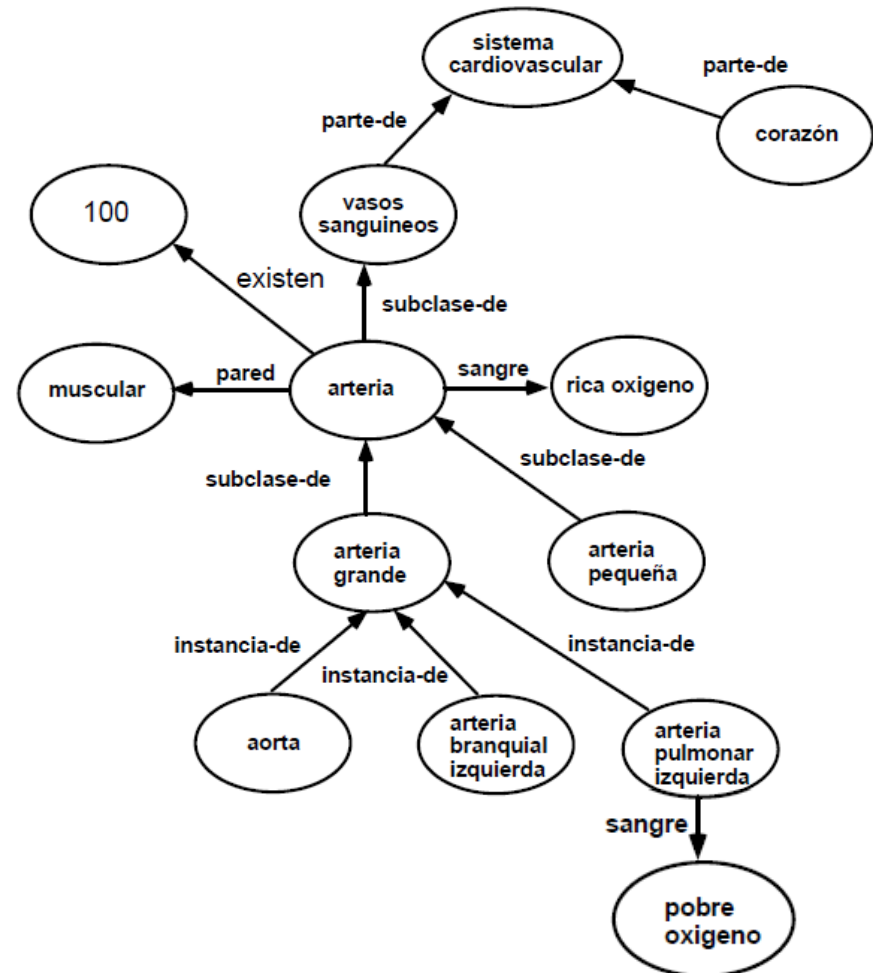
La propiedad "las arterias transportan sangre rica en oxígeno" no debe ser heredada (excepción) por la arteria pulmonar izquierda.

Una posible solución es:

- almacenar la propiedad como información explícita en cada concepto en el que se cumple la propiedad, eliminando la propiedad general.

b) No heredar propiedades que consideramos relevantes para una clase, pero no para sus especializaciones.

"Existen 100 arterias"

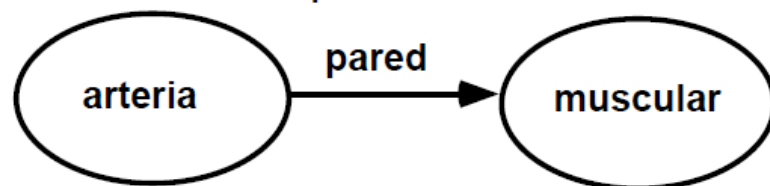




# Redes semánticas extendidas

- ◆ Las Redes Semánticas Extendidas (A. Deliyanni y R. A. Kowalski): formalismo de representación alternativo a la forma clausal de la lógica con la restricción de solo poder utilizar símbolos de predicado binarios.
- ◆ Debido a la equivalencia sintáctica entre redes semánticas extendidas y la forma clausal de la lógica, las reglas de inferencia definidas para la forma clausal de la lógica pueden ser aplicadas para manipular arcos y nodos de una red semántica extendida.
- ◆ Un predicado binario puede ser traducido en una red en la que:

- los nodos representan términos
- el arco representa la relación (predicado)



pared (arteria, muscular)

- ◆ La restricción a símbolos de predicado binarios no es crítica, ya que cualquier átomo que contenga un símbolo de predicado n-ario puede ser reemplazado por una conjunción de átomos que contengan solo símbolos de predicado binarios.

Si  $n > 2$  se requieren  $n + 1$  nuevos predicados.

Si  $n = 1$ , solo se requiere un nuevo predicado.

# Redes semánticas extendidas

## Ejemplo:

PresiónSangre(x, y, z) = “la presión sanguínea de x varia entre y mmHg y z mmHg”

El predicado

PresiónSangre (arteria, 40, 80)

puede ser reemplazado por la conjunción de predicados binarios:

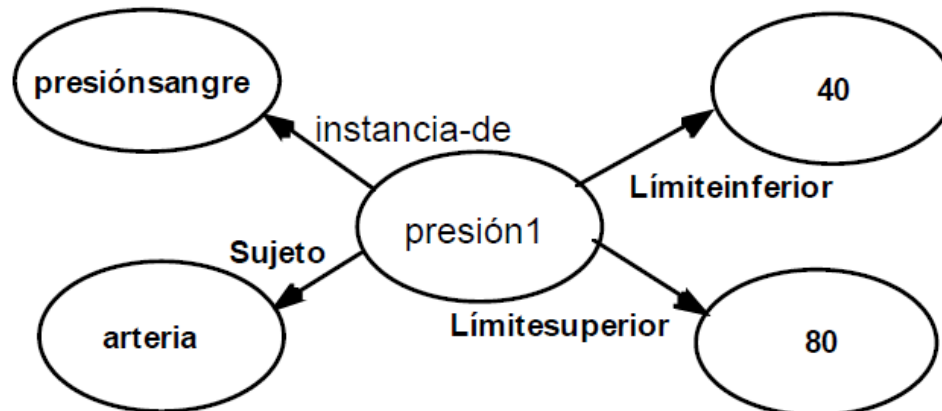
instancia-de(presión1, presiónsangre)

sujeto(presión1, arteria)

Límiteinferior(presión1, 40)

Límitesuperior(presión1, 80)

Red semántica equivalente:



# Redes semánticas extendidas

**Ejemplo:** Traducción predicado unario a binario

Supongamos el siguiente predicado unario:

$\text{Arteria}(x) = \text{"x es una arteria"}$

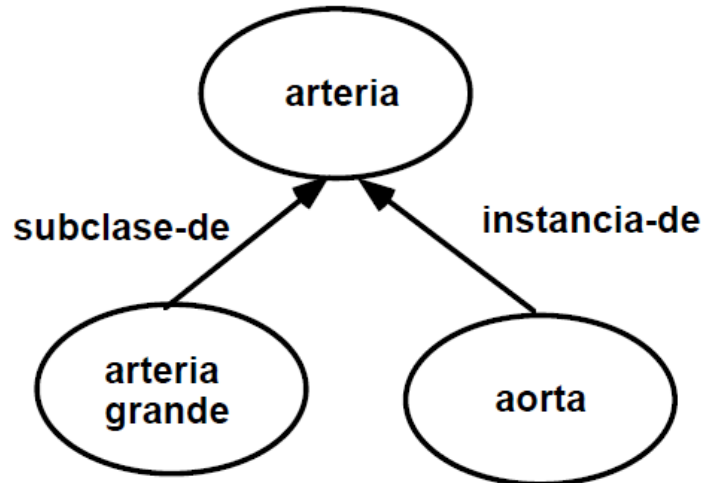
y las cláusulas

$\text{Arteria}(\text{aorta})$  y  $\text{Arteria}(\text{arteria-grande})$

Estas cláusulas pueden ser reemplazadas por las cláusulas

$\text{instancia-de}(\text{aorta}, \text{arteria})$

$\text{subclase-de}(\text{arteria-grande}, \text{arteria})$



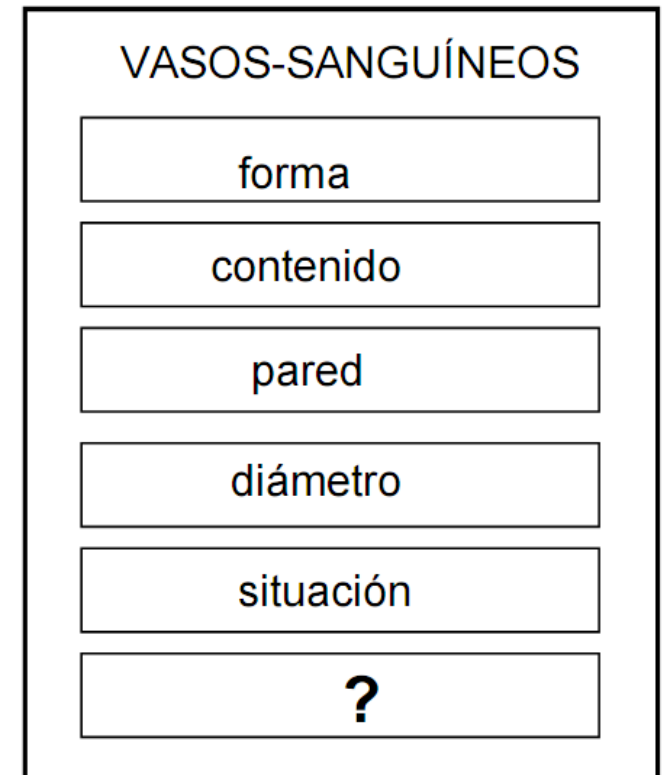
# Frames

- Frames (Minsky, 1975)
  - se basan en el concepto de considerar la resolución de problemas humana como el proceso de rellenar huecos de descripciones parcialmente realizadas (O. Selz)
  - la idea subyacente en un sistema basado en frames es que el conocimiento concerniente a individuos o clases de individuos, incluyendo las relaciones entre los mismos, es almacenada en una entidad compleja de representación llamada frame (unidad,objeto,concepto)
  - un conjunto de frames que representa el conocimiento de un dominio de interés es organizada jerárquicamente en lo que es llamado una taxonomía (asociada a un método de razonamiento automático llamadoherencia).

# Frames

- El conocimiento relevante de un concepto (objeto individual o clase de objetos) es representado mediante entidad compleja de representación llamada frame, constituida por un conjunto de propiedades (atributos)
- Las frame proporcionan un formalismo para agrupar explícitamente todo el conocimiento concerniente a las propiedades de objetos individuales o clases de objetos.
- Tipos de frames:
  - frames clase, o frames genéricas, que representan conocimiento de clases de objetos
  - frames instancia, representan conocimiento de objetos individuales.

## FRAME



# Jerarquía de frames

◆ El conocimiento de un dominio de interés es organizado jerárquicamente en una jerarquía o taxonomía de frames.

◆ La taxonomía es representada mediante un grafo dirigido acíclico (generalmente un árbol) en el que solo se dan las relaciones:

- instancia-de
- subclase-de

donde cada nodo denota una frame.

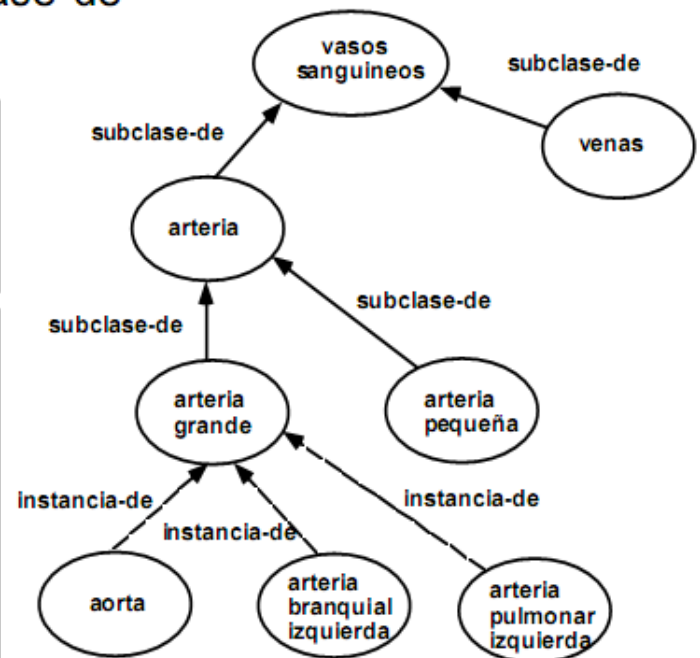
- raíz del árbol: descripción más general del dominio.

- hojas del árbol: descripciones de conceptos más específicos.

- especializaciones (instancias, subclases, subframe) descendiente de una frame en la taxonomía.

no se pueden definir especializaciones de las frames instancia (excepción metaclases).

- generalizaciones (superclases, superframe) antecesores de una frame en la taxonomía.



**Las propiedades de las frame más generales son heredadas por sus especializaciones (herencia).**

# Definición de frame

## FRAME

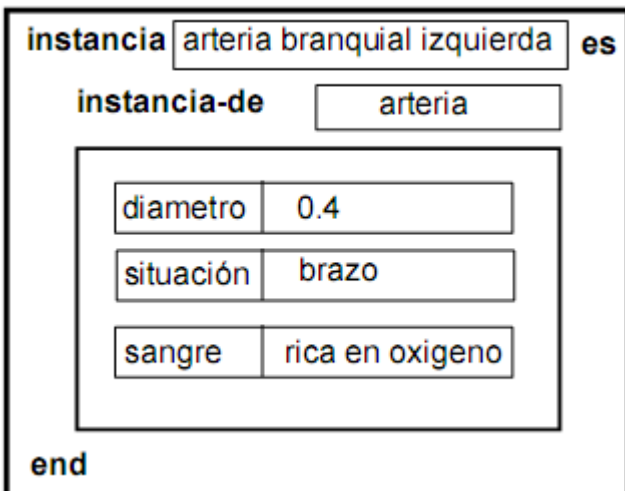
clase | instancia  es

subclase-de | instancia-de

atributo	valor   tipo
atributo	valor   tipo
atributo	valor   tipo
⋮	
atributo	valor   tipo

end

- cada frame de una taxonomía tiene un nombre único.
- una frame solo puede tener una superclase (herencia simple).
- la información (propiedades) específica al concepto representado por una frame es representada mediante atributos o slots
- los atributos ofrecen un medio de representar las propiedades de objetos individuales o clases de objetos.



# Sintaxis de frames

```

<frame> ::=
    <clase> | <instancia>

<clase> ::=
    clase <nombre-clase> es
        subclase-de <espec-super>;
        <atributos-clase>
    end

<instancia> ::=
    instancia <nombre-instancia> es
        instancia-de <espec-super>;
        <atributos-instancia>
    end

<espec-super> ::=
    <nombre-clase> | nil

<atributos-clase> ::=
    <declaración> {; <declaración>}* | <vacío>

<atributos-instancia> ::=
    <par-atributo-valor> {; <par-atributo-valor>}* | <vacío>

<declaración> ::=
    <par-atributo-tipo> | <par-atributo-valor>

<par-atributo-tipo> ::=
    <nombre-atributo> : <tipo>

<par-atributo-valor> ::=
    <nombre-atributo> = <valor>

<tipo> ::=
    entero | real | string | <conjunto> | <nombre-clase>

<valor> ::=
    <constante> | <nombre-instancia>
    <vacío> ::=

```

- el símbolo nil denota que una frame es la raíz de la taxonomía.
- <conjunto> denotará un conjunto enumerado de constantes elementales y/o nombres de instancias
- Se asume que un par atributo/tipo o atributo/valor, ocurre una única vez en una taxonomía (posteriormente se elimina esta suposición).



# Ejemplos

```
instancia arteria-branquial-izquierda es  
  instancia-de arteria;  
    diámetro = 0.4;  
    situación = brazo;  
    sangre = rica-oxigeno  
end  
instancia brazo es  
  instancia-de miembro;  
    posición = superior  
end
```

```
clase arteria es  
  subclase-de vasos-sanguíneos;  
    situación: {brazo, cabeza,  
               pierna, tronco}  
end  
clase miembro es  
  subclase-de nil;  
    posición:{superior, inferior}  
end
```

- la información especificada en las partes atributo de las frames instancia sigue las siguientes reglas:
  - todos los atributos que ocurren en las instancias de una frame clase deben haber sido declarados en dicha frame clase o en una de sus generalizaciones.
  - los valores asignados a los atributos de la instancia deben ser del tipo de datos definido en alguna de sus generalizaciones.

# Ejemplos

◆ en la declaración de una frame clase también se puede asignar valor a un atributo (están permitidos los pares atributo valor)

```
clase arteria es  
  subclase-de vasos-sanguíneos;  
  situación : {brazo, cabeza,  
               pierna, tronco};  
  sangre = rica-oxigeno  
end
```

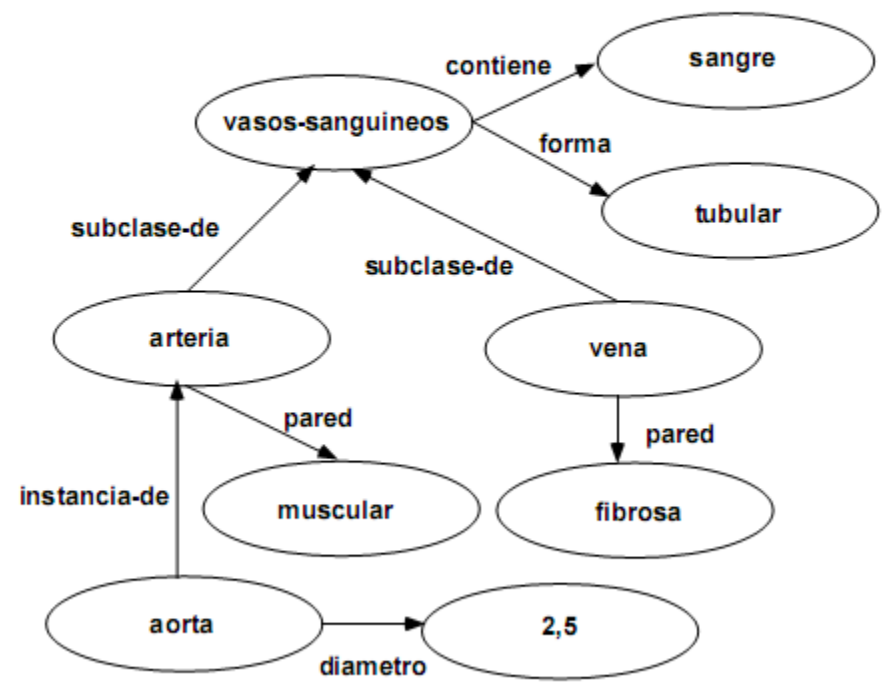
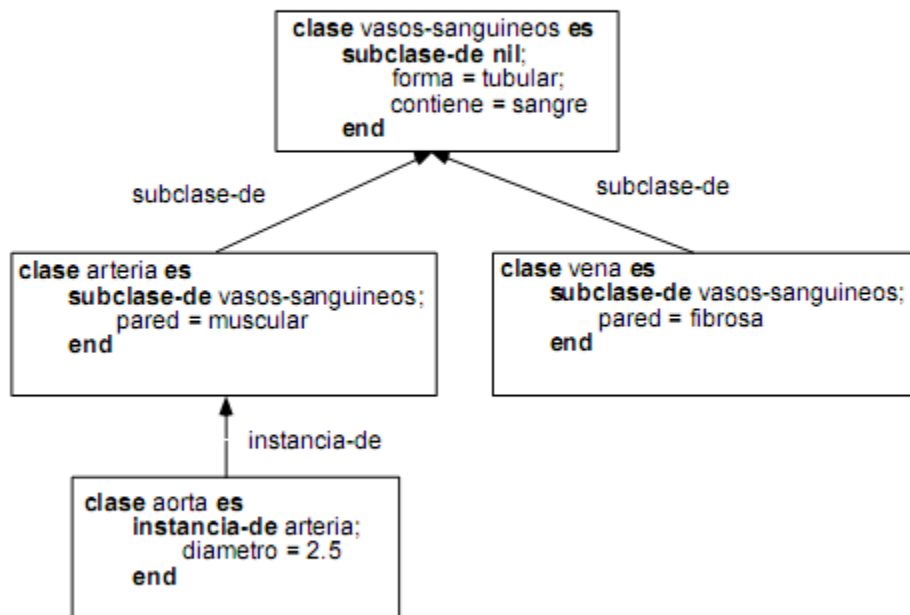
```
instancia arteria-branquial-izquierda es  
  instancia-de arteria;  
  diámetro = 0.4;  
  situación = brazo  
end
```

“ se considera adecuado especificar el valor rica-oxigeno del atributo sangre en la especificación de la clase, con lo que no es necesario especificarlo en sus subclases o instancias, ya que esta propiedad será heredada por todos los descendientes de la clase”

# Equivalencias Frames / Redes Semánticas

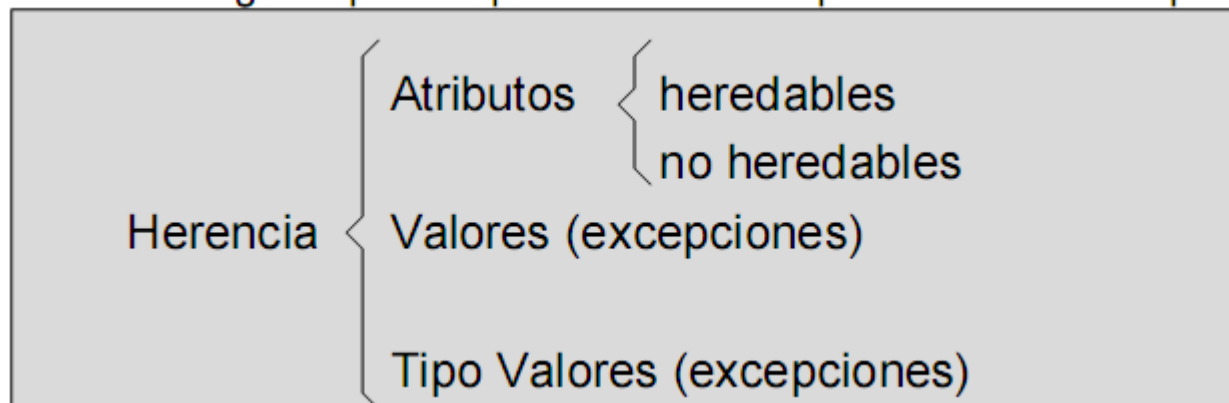
clases, instancias y valores atributos ≡ conceptos (nodos)

atributos ≡ relaciones (arcos)



# Herencia Simple

- Herencia simple (frames con una única superclase, taxonomías de tipo árbol).
- Herencia múltiple (frame con más de una superclase, taxonomías de tipo grafo).
- Tipos de información sobre atributos en una taxonomía de frames:
  - información sobre tipo de atributo
  - información sobre valor de atributo
- Los enlaces instancia-de y subclase-de definen una ordenación parcial de frames clase en la taxonomía, que puede ser utilizado para razonar sobre los valores de los atributos de la misma forma que en las redes semánticas (las especializaciones 'heredan' los valores de los atributos de las generalizaciones).
- El enlace subclase-de puede ser considerado como una relación que restringe los contenidos semánticos en la taxonomía de frames (los valores de un atributo de una subclase están restringidos por el tipo del atributo especificado en la superclase).



# Herencia simple de atributos/valores

**La herencia simple consiste en que una frame hereda todos los atributos de sus superclases, así como los valores de estos atributos.**

- Consideremos lo siguiente desde un punto de vista de la lógica de predicados de primer orden:

$$\{\text{arteria}(\text{aorta}), \forall x(\text{arteria}(x) \rightarrow \text{pared}(x)=\text{muscular})\}$$

aplicando el Modus Ponens obtenemos:

$$\text{pared}(\text{aorta}) = \text{muscular}$$

de forma similar derivaríamos:

$$\text{vasos-sanguíneos}(\text{aorta})$$

*Esta información es heredada por la 'aorta' desde una información general de las arterias.*

- Este tipo de razonamiento es modelado en un método de inferencia para frames llamado 'herencia simple'. Informalmente podemos definir el procedimiento de herencia simple de la siguiente forma:
  - *Recorrer la taxonomía desde una frame específica hasta la raíz de la misma y coleccionar sucesivamente los atributos de las frames encontradas y sus valores asociados*

# Excepciones de la herencia simple

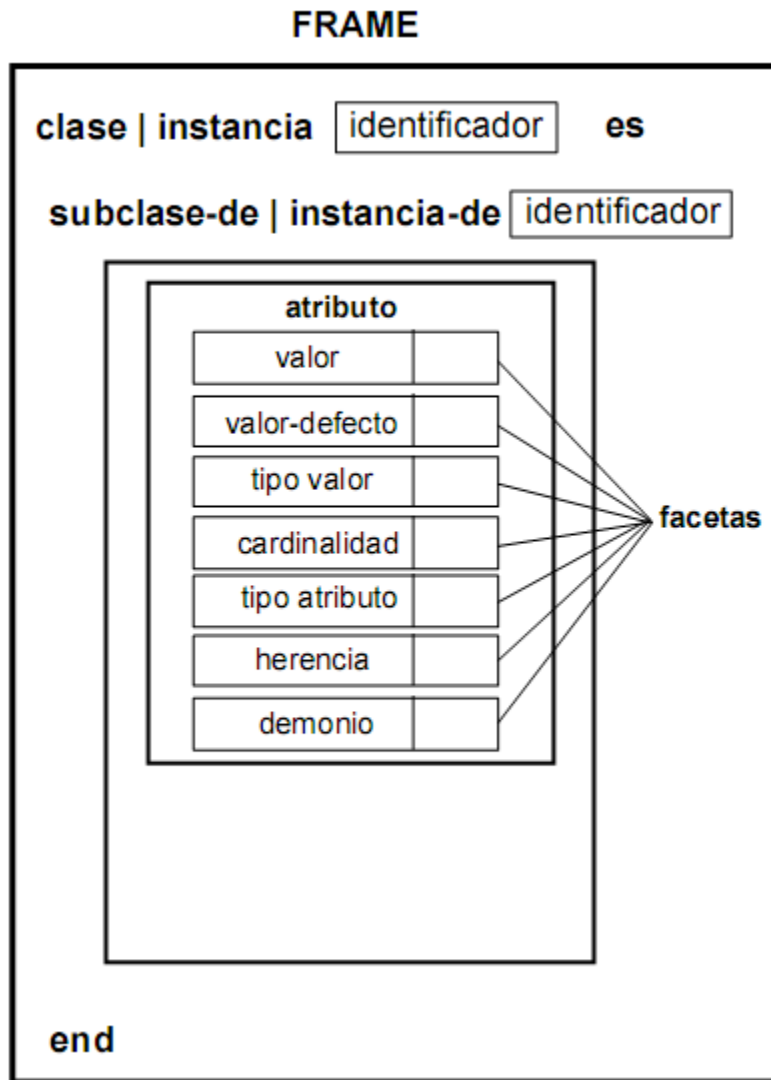
- unicidad atributos en taxonomía  $\Rightarrow$  disminuye expresividad.
- asumiremos atributos mono-valor.
- un atributo puede ocurrir en más de una frame.

```
clase arteria es  
  subclase-de vasos-sanguíneos;  
  sangre = rica-oxigeno  
end
```

```
instancia arteria-pulmonar-izquierda es  
  instancia-de arteria;  
  sangre = pobre-oxigeno;  
end
```

- Solución: concepto de ‘excepción’ (propiedad general que no se cumple para algún(os) objeto(s) del dominio).
  - Si en la especificación de una subclase o instancia, se especifica un valor de un atributo que ha sido especificado también en una generalización de la misma, se mantiene dicho valor y no se hereda el valor especificado en las generalizaciones (los descendientes de esta nueva frame heredaran el nuevo valor especificado como excepción)”

# Extensión de la definición de frame



◆ El formalismo de frames descrito no permite:

- saber si el valor del atributo de una instancia ha sido heredado o ha sido especificado explícitamente.
- calcular los valores de un atributo a partir de los valores de otros atributos.

◆ Muchos lenguajes de frames proporcionan constructores especiales del lenguaje llamados facetas, que permiten manejar las funcionalidades anteriores.

# Extensión de la definición de frame

- ◆ Una **faceta** es considerada como una propiedad asociada a un atributo.
  - **faceta valor**, es la más común y referencia el valor real del atributo.
  - **faceta valor por defecto**, denota el valor inicial del atributo en caso de que no se especifique lo contrario.
  - **faceta tipo valor**, especifica el tipo de datos del valor del atributo.
  - **faceta cardinalidad**, especifica si se trata de un atributo uni o multi-valuado.
  - **faceta máxima cardinalidad**, solo es valida para atributos multi-valuados y especifica el máximo número de valores asociados al atributo.
  - **facetas demonio**, permiten la integración de conocimiento declarativo y procedural. Un demonio o valor activo es un procedimiento que es invocado en un momento determinado durante la manipulación del atributo donde ha sido especificado (si-necesario, si-añadido, si-eliminado).
  - **faceta tipo atributo**, especifica si se trata de un atributo heredable o no heredable.
  - **faceta herencia**, especifica el tipo de herencia del atributo.



# Sintaxis frames

( *extensión del lenguajes de frames* )

<espec-super>::=	<nombre-clase>   <b>nil</b>
<atributos>::=	<atributo-faceta-par>   {; <atributo-faceta-par>*   <vacío>
<atributo-faceta-par>::=	<nombre-atributo>=( <espec-faceta>   {, <-espec-faceta>* )
<espec-faceta>::=	<faceta>   <b>demonio</b> <tipo-demonio> <llamada-demonio>
<faceta>::=	valor <valor>   defecto <valor>   tipo-valor : <tipo>   max-card <b>entero</b>   card <tipo-card>   herencia <tipo-herencia>        tipo-atributo <tipo-atributo>
<tipo-demonio>::=	<b>si-necesario</b>   <b>si-añadido</b>   <b>si-eliminado</b>
<tipo>::=	<b>entero</b>   <b>real</b>   <b>string</b>   <conjunto>   <nombre-clase>   <rango>
<valor>::=	<constante>   <nombre-instancia>
<vacío>::=	
<tipo-atributo> ::=	<b>heredable</b>   <b>no_heredable</b>
<tipo-herencia>::=	<b>Z</b>   <b>N</b>
<tipo-card> ::=	<b>multi</b>   <b>uni</b>
<conjunto> ::=	{ <valor>   {, <valor>* }
<rango> ::=	[<constante> .. <constante>]

# Ejemplo

```
clase arteria es  
  subclase-de vasos-sanguíneos;  
    pared = (valor muscular);  
    sangre =(defecto rica-oxigeno, tipo-valor:{rica-oxigeno, pobre-oxigeno});  
    presión-sanguínea = (defecto 20);  
    flujo-sanguíneo =(defecto 4);  
    resistencia =(demonio  
                  si-necesario  
                  R(presión-sanguínea,flujo-sanguíneo))  
end
```

## **Criterios de herencia: N-herencia**

### **Prioridad a las facetas de los atributos**

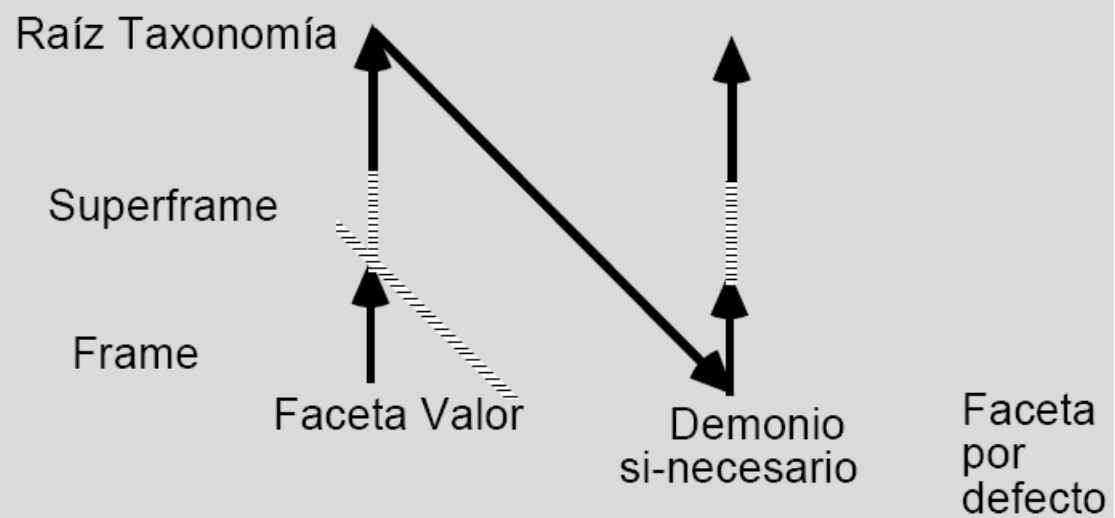
- 1. Consultar la faceta valor del atributo en el marco considerado.**
- 2. Si no tiene valor, subir un nivel en la taxonomía y consultar valor.**
- 3. Repetir la operación hasta encontrar valor en algún marco.**
- 4. Si tampoco hay éxito, consultar función-para-calcular-valor del atributo**
- 5. Si el paso 3 no tiene éxito, volver al marco considerado y considerar valor-por-defecto del atributo y repetir la búsqueda hacia arriba**

## **Criterios de herencia: N-herencia**

### **Prioridad a las facetas de los atributos**

- 1. Consultar la faceta valor del atributo en el marco considerado.**
- 2. Si no tiene valor, subir un nivel en la taxonomía y consultar valor.**
- 3. Repetir la operación hasta encontrar valor en algún marco.**
- 4. Si tampoco hay éxito, consultar función-para-calcular-valor del atributo**
- 5. Si el paso 3 no tiene éxito, volver al marco considerado y considerar valor-por-defecto del atributo y repetir la búsqueda hacia arriba**

## N-herencia



## **Criterios de herencia: Z-herencia**

**Prioridad a los marcos más cercanos:**

- 1. Consultar la faceta valor del atributo en el marco considerado.**
- 2. Si tampoco tiene, consultar función-para-calcular-valor.**
- 3. Si no tiene valor, consultar valor-por-defecto.**
- 4. Agotadas las posibilidades del marco considerado, se sube en la jerarquía y se hace lo mismo con el marco padre**

## Z-herencia

Raíz Taxonomía

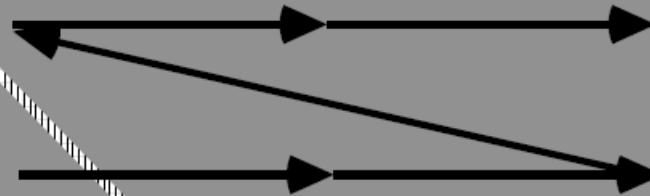
Superframe

Frame

Faceta Valor

Demonio  
si-necesario

Faceta  
por  
defecto



# Herencia de tipo de Atributos

**relación supertipo**

**relación subtipo**

- ◆ una especialización hereda el tipo de valor de su generalización a no ser que se especifique lo contrario.
- ◆ excepciones: en la especificación de una frame se puede restringir el tipo de valor de un atributo a un subtipo del tipo de valores de su generalización

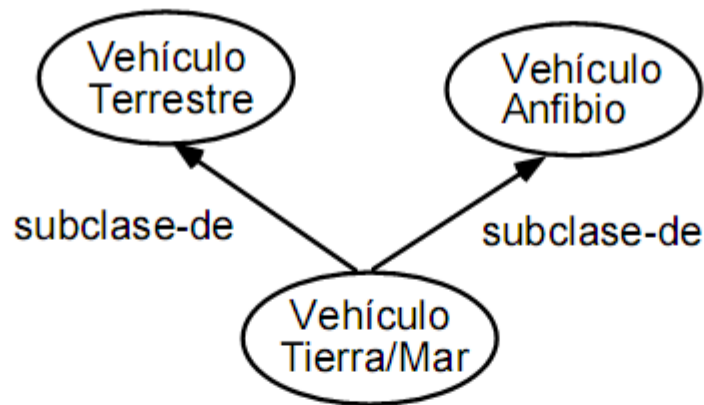
```
clase vasos-sanguíneos es  
  subclase-de nil;  
  sangre = (tipo-valor {rica-oxigeno, pobre-oxigeno});  
  pared = (tipo-valor {muscular, fibrosa, mixta})  
end  
  
clase arteria es  
  subclase-de vasos-sanguíneos;  
  pared = (tipo-valor {muscular, mixta});  
  espesor-pared = (tipo-valor real)  
end
```

- ◆ tipo de valor heterogéneo

**<tipo> ::= entero | real | string | <conjunto> | <nombre-clase> | <tipo> {U <tipo>}\***



# Herencia Múltiple



- ◆ taxonomía representada por un grafo dirigido aciclico.

```
<clase> ::= clase <nombre-clase> es  
           subclase-de <espec-super> {, <espec-super>*}; <atributos>  
           end
```

- ◆ una especificación hereda los atributos de todas sus generalizaciones (su conjunto de atributos será la unión de los conjuntos de atributos de sus superclases).
- ◆ excepciones:
  - debidas a inconsistencias entre generalizaciones y especializaciones se resuelven mediante herencia simple.
  - debidas a inconsistencias entre superclases de una misma especialización, se necesitan métodos para decidir que valores de facetas heredar de entre los de las superclases.
  - explicitas

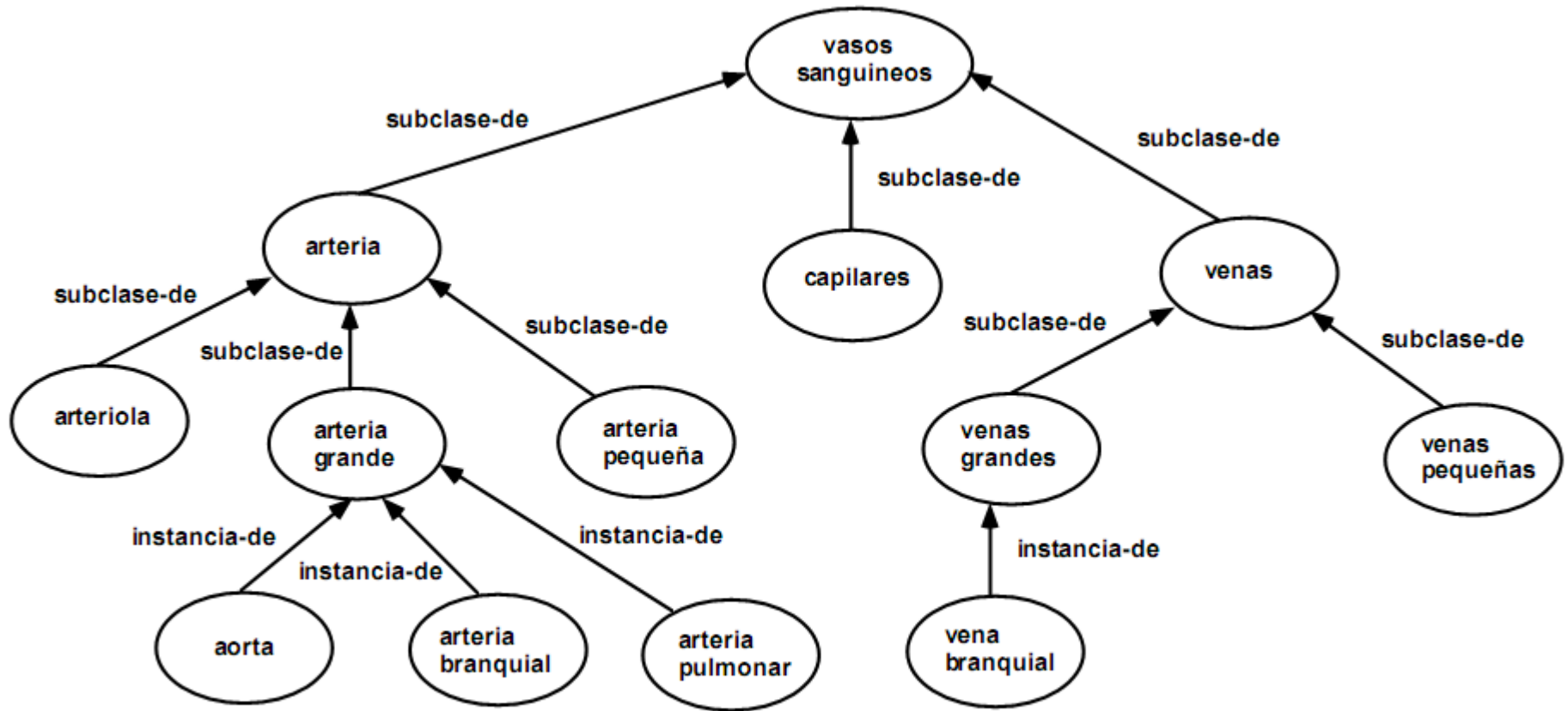
# Ejemplo

a) Representar mediante un método basado en frames, detallando:

- Clases, subclases e instancias.
- Slots o atributos de las frames. Distinguir miembros y propios.
- Clase de valores de los atributos.
- Valores de los atributos, para aquellos que sean conocidos.

Categoría	Rango de presión media (mmHg)
Arterias grandes	90-100
Arterias pequeñas	80-90
Arteriolas	40-80
Venas	<10
Arteriolas	<10

# Ejemplo



# Ejemplo

clase vasos-sanguíneos es

    subclase-de nil;

        pared = (tipo-valor {muscular, fibroso}, tipo-atributo heredable);

        oxigeno = (tipo-valor {rico, pobre}, tipo-atributo heredable);

        presión-media = (tipo-valor real, tipo-atributo heredable,  
                            demonio si-necesario

                            media(presión-máxima, presión-mínima));

        presión-mínima = (tipo-valor real, tipo-atributo heredable);

        presión-máxima = (tipo-valor real, tipo-atributo heredable)

end

clase arterias es

    subclase-de vasos-sanguíneos;

        pared = (valor muscular);

end

# Ejemplo

clase capilares es

    subclase-de vasos-sanguíneos;

        presión-media = (tipo-valor [0..10])

end

clase arterias-grandes es

    subclase-de arterias;

        presión-media = (tipo-valor [90..100])

end

clase arterias-pequeñas es

    subclase-de arterias;

        oxigeno = (valor rico);

        presión-media = (tipo-valor [80..90])

end

# Ejemplo

clase arteriolas es

    subclase-de arterias;

        oxigeno = (valor rico);

        presión-media = (tipo-valor [40..80])

end

clase venas es

    subclase-de vasos-sanguíneos;

        pared = (valor fibroso);

        oxigeno = (valor pobre);

        presión-media = (tipo-valor [0..10])

end