

### 6.1 Introducción

Hay dos técnicas para formalizar conocimiento (**representación** y **razonamiento**).

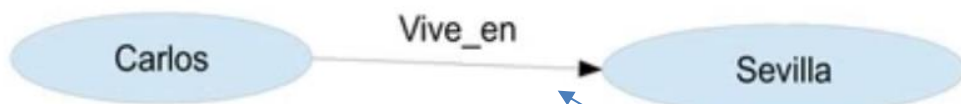
Tipos de formalismos:

- Basados en **conceptos**: Representan posibles clases o entidades de dominio. **Marcos**
- Basados en **relaciones**: Relaciones entre conceptos o entidades. **Lógica, redes semánticas**
- Basados en **acciones**: Conocimiento cómo acciones básicas. **Reglas, guiones...**

### 6.2 Redes semánticas

Son grafos orientados formados por:

- **Nodos** etiquetados: **conceptos** e **instancias**
- **Arcos unidireccionales** etiquetados: **relaciones**



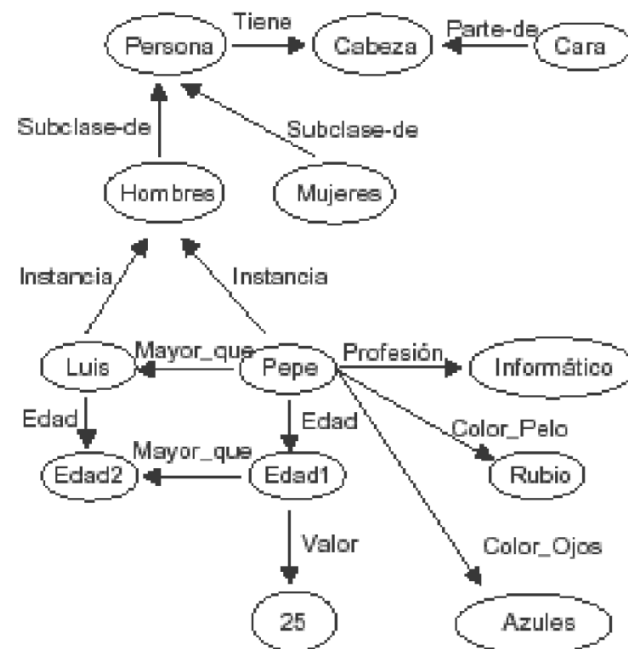
Hay **dos tipos de arcos**:

- **Descriptivos**: que describen entidades y conceptos (vive\_en, profesión...) la semántica depende del dominio.
- **Estructurales**: enlazan entidades y conceptos formando la arquitectura de red. Tienen semántica independiente del dominio, las más habituales:
  - **Generalización**: Relaciona una clase con otra más general (subclase).
  - **Instancia**: Liga un objeto con su tipo genérico (instancia).
  - **Agregación**: Liga un objeto con sus componentes (parte\_de).

**Por ejemplo**: Explique los diferentes tipos de arcos que pueden aparecer en una red semántica. Ilustre la explicación con un ejemplo tomado del mundo real.

Los arcos estructurales marcan la arquitectura de la red, son genéricos, no dependen del objeto que se está modelando.

Los descriptivos son propiedades que tienen que ver con el conocimiento modelado.



Otro concepto importante es la **aridad**:

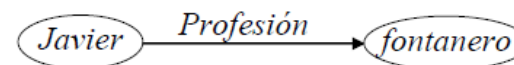
En este caso, relacionamos a Carlos con Sevilla por medio de la relación **vive\_en**. En este caso, el predicado tiene aridad 2 (tiene dos argumentos) **Propiedad(X, Y): Vive\_en(Carlos, Sevilla)**.

Para representar predicados no binarios:

- **Aridad uno**: Hombre(Pepe) se representa cómo INSTANCIA(Pepe, Hombre).
- **Aridad mayor a dos**: Se crea un nuevo objeto que representa el predicado y nuevos predicados binarios expresan las relaciones con los argumentos.

**Por ejemplo**: Describa cómo se pueden representar predicados no binarios mediante redes semánticas. Ponga un ejemplo que ilustre su descripción.

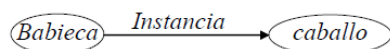
La etiqueta asociada a un arco de una red semántica suele representar un predicado binario cuyo par de argumentos son el objeto origen y el objeto destino unidos por el arco. Por ejemplo, el hecho Profesión(Javier, fontanero), que consta del predicado binario "Profesión", se puede representar mediante la siguiente red semántica:



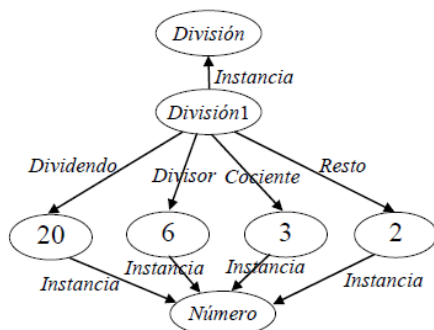
## Fundamentos de inteligencia artificial

Profesión Pero también es posible representar predicados no binarios mediante redes Y ahí se relacionan un montón de frases nominales (Pedro, playa, autobús, sábado) y semánticas. Por ejemplo, el hecho Caballo(Babieca) es equivalente a Instancia(Babieca, cada uno tiene un caso (papel) Pedro es el agente, playa es el destino, autobús el Caballo). Por tanto, en este ejemplo el predicado unario "Caballo" se representaría del instrumento...

siguiente modo mediante una red semántica:



Instancia Si se quisieran representar predicados de aridad mayor que dos, habría que operar de un modo diferente. En este caso, el predicado de aridad mayor que dos se representa mediante un objeto y se utilizan predicados binarios adicionales para describir las relaciones entre este objeto y sus argumentos. A modo de ejemplo, si consideramos el hecho División(20, 6, 3, 2), que representa la operación de división de 20 entre 6 cuyo cociente es 3 y cuyo resto es 2, la red semántica que lo representa sería:



Como puede comprobarse en la figura, se ha creado un nodo que representa el predicado no binario (División1) y cuatro predicados binarios (Dividendo, Divisor, Cociente y Resto) que representan las relaciones con los cuatro argumentos del predicado no binario.

Se pueden representar oraciones simples o compuestas que describen acciones.

**Gramática de casos:** verbo principal y frases nominales que se relacionan con el verbo a través de un caso. Casos: agente, objeto, resultado, instrumento, origen, destino.

Por ejemplo: Pedro fue a la playa en autobús el domingo.

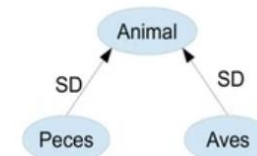
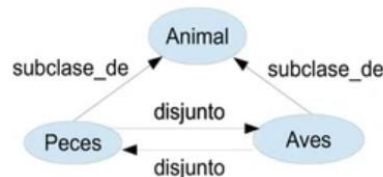
El verbo principal es ir, en modo pasado.

Los verbos pueden también estar afectados por modalidades:

- **Tiempo** (presente, pasado, futuro)
- **Voz** (activa, pasiva)

Hay veces que se hace necesario especificar qué entidades de dominio son disjuntas (Por ejemplo, usar la etiqueta disjunto).

O usar diferentes etiquetas estructurales: **S** (Subconjunto), **SD** (subconjunto disjunto), **E** (elemento), **ED** (elemento disjunto).

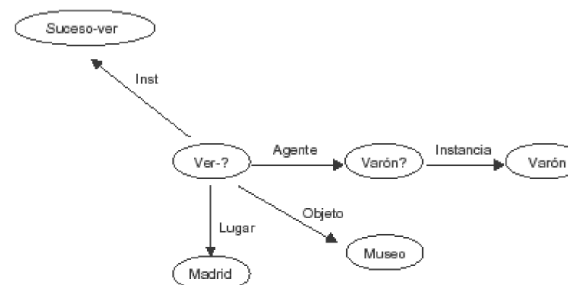


### 6.3 Inferencias de conocimiento en redes semánticas

Se busca usar el conocimiento en una red semántica para responder preguntas que se planteen.

**Equiparación:** Un fragmento (apunte) de red (por ejemplo, una consulta), se equipara con una red semántica si se puede asociar a un fragmento de la red:

1. Se construye un **apunte** que responde a la pregunta propuesta. Se usan **nodos constantes** (valores conocidos), **nodos variables** (datos preguntados) y **arcos etiquetados**.
2. Se **superpone** el apunte sobre la red semántica
3. Los nodos variables se ligán a los nodos constantes de la red hasta encontrar una equiparación perfecta.
4. La **respuesta** a la consulta es el **fragmento** de la red con los **valores** que rellenan los nodos variables. Si no hay equiparación posible, la respuesta es que no se conocen los datos pedidos.



# Fundamentos de inteligencia artificial

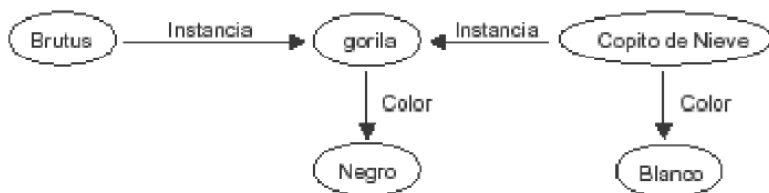
**Por ejemplo:** Describa detalladamente las técnicas de inferencia en redes semánticas.

En cuanto a la técnica de equiparación:

- **Definición:** Tomar un trozo de una red semántica con constantes y variables para intentar dar respuesta a una pregunta que se ha lanzado usando el conocimiento que hay en la red.
- **Pasos:** construcción de un apunte (o fragmento de red), cotejo del apunte con la red semántica de la base de conocimientos, ligadura de los nodos variables del apunte con nodos constantes de la red semántica y, finalmente, devolución (como resultado) del apunte con los valores ligados a los nodos variables.

## Herencia:

- Permite que nodos específicos de una red accedan a propiedades definidas en otros nodos siguiendo los arcos **instancia** y **subclase\_de**
- Se hace una exploración de la red partiendo del nodo consultado y siguiendo esos enlaces (anchura, profundidad) hasta encontrar el valor del atributo.
- **Razonamiento no monótono:** las propiedades pueden presentar **excepciones** en sus valores. Se hereda entonces el **valor de la propiedad más cercano** al nodo consultado.



**Por ejemplo:** Describa detalladamente las técnicas de inferencia en redes semánticas.

En cuanto a la técnica de herencia de propiedades, deben desarrollarse los siguientes aspectos:

- **Definición:** Pedir una propiedad de una instancia o clase más específica, pero que no está definida en esa clase pero que si se puede encontrar en otra más general.
- **Tratamiento de excepciones:** se hereda el valor de la propiedad del nodo más cercano al nodo que sirvió como punto de partida a la inferencia.

## 6.4 Marcos

Son grafos dirigidos en cuyos nodos se estructura la información correspondiente a una entidad del dominio.

- Es la técnica de representación de conocimientos más utilizada en IA.
- Unen el conocimiento declarativo y procedimental

Sus recursos para presentar el conocimiento son:

- **Marcos:** **Conceptos** y **elementos** (instancias)
- **Relaciones:** **Dependencias** entre conceptos

- **Propiedades:** **Describen** cada concepto

- **Facetas:** **Posibles valores** de cada propiedad.

**Tipos de marcos:** Hay dos tipos:

- **Marcos clase:** Conceptos, clases o situaciones genéricas descritos por un conjunto de propiedades. Cada propiedad puede o no tener un valor
- **Marcos instancia:** Individuos. Han de estar relacionados al menos con un marco clase. Algunas propiedades se heredan, otras tienen valores específicos.

**Relaciones entre conceptos:**

- **Estándar:** Son **instancias** (Desde marco instancia a marco clase) y **subclase\_de** (entre dos clases). Sus inversas, respectivamente **representa** y **superclase\_de**. Sirven para la herencia.
- **No estándar:** Dependencias entre conceptos. Por ejemplo: fraternal (entre clases que son subclase del mismo padre), disjuntos...
- **A medida o "ad hoc":** Siempre entre marcos clases. Si se usan en marcos instancia, han de estar definidas para sus clases correspondientes.

**Por ejemplo:** Explique los diferentes tipos de relaciones que pueden aparecer en un sistema de marcos. Ilustre la explicación con ejemplos tomados del mundo real.

Existen dos tipos de marcos, clases e instancias, y mientras que las primeras expresan conceptos genéricos, las segundas constituyen ejemplos concretos de dichos conceptos genéricos. Los diferentes marcos quedan enlazados entre sí por medio de relaciones.

Las relaciones estándar más importantes son las de "Subclase-de", que une una clase específica con otra clase más genérica que la engloba, y la de "Instancia", que une una instancia con la clase a la que pertenece. Sobre este tipo de relaciones se puede aplicar herencia de propiedades.

Existen otras relaciones no estándar que expresan dependencias entre conceptos del dominio y sobre las que no se puede aplicar herencia de propiedades. Por ejemplo, la relación "Fraternal" une dos clases cuya clase padre coincide, la relación "Disjunta" une dos clases que no pueden tener una instancia común y, finalmente, diferentes relaciones "A medida" o "Ad-hoc" entre dos clases pueden ser definidas por el usuario. Las relaciones ad-hoc sólo se pueden definir entre marcos clase y para que se puedan definir también entre marcos instancia previamente hay que comprobar que: (1) La relación ad-hoc debe haber sido definida previamente entre dos marcos clase y (2) Los marcos instancia deben ser respectivamente instancias de dichos marcos clase.

# Fundamentos de inteligencia artificial

Algunos de los entornos para construir sistemas basados en marcos no implementan relaciones no estándar. En este caso, el ingeniero de conocimiento las representaría mediante marcos adicionales y haría inferencia con ellas mediante: procedimientos, reglas o demonios.

## Propiedades entre conceptos:

Describen conceptos. Los hay de dos tipos:

- **Propiedades de clase:** Con características genéricas de un concepto o clase. Se definen y rellenan en un marco clase y tienen el mismo valor en todas las instancias.
- **Propiedades de instancia:** Se definen en un marco clase, pero se rellenan en cada instancia.

## Facetas de propiedades:

Modelan características de las propiedades y relaciones de los marcos clase. Mantienen la integridad semántica de los datos. Los hay de:

- **Tipo ranura:** Indican tipo de datos para propiedad o relación
- **Cardinalidad mínima/máxima y multivaluada:** Número de valores que puede tomar
- **Valores permitidos:** Conjunto de valores posibles
- **Valores por omisión:** Son los valores que toma la propiedad cuando no se especifica
- **Si necesito / si modifico / si añadido / si borro:** Facetas que almacenan procedimientos que se ejecutan al consultar o cambiar el valor de una propiedad de instancia en un marco de instancia.

## 6.5 Inferencia de conocimiento en SBM.

Las técnicas más usadas son:

**Equiparación:** Para clasificar entidades en la base de conocimientos. Tiene tres etapas que se inician con un marco pregunta:

1. Selección de **marcos candidatos**. Puede haber varios. Ya que las propiedades de la nueva entidad pueden estar en distintos marcos. O ser desconocidas.
2. Cálculo del **valor de equiparación** para cada candidato (Valor que indica la calidad de la equiparación)
3. **Decisión** de los marcos de clases con que se equipara la nueva entidad

## Herencia de propiedades:

Para compartir propiedades entre marcos

Consulta el valor de una propiedad en una instancia.

**Herencia simple:** Solo hay un camino desde el nodo raíz hasta la instancia. Se devuelve el valor de la propiedad indicado en la instancia o el nodo más cercado a él, siguiendo enlaces **instancia** y **subclase\_de**.

**Herencia múltiple:** Hay varios caminos de la raíz hasta E. (Un hijo puede heredar propiedades de varios padres, para saber de cuál toma el valor).

**Búsqueda en profundidad:** Con criterios para evitar repeticiones, se puede ordenar topológicamente para facilitar el trabajo.

**Búsqueda en amplitud (longitud del camino)** Puede ser ambiguo.

**Distancia inferencial:** La clase A está más cercada a B que C, si hay un camino de A a C que pasa por B.

## Valores activos (demonios):

Representan la conducta del sistema y mantienen la integridad de los datos almacenados

- Son un componente procedimental de los marcos.
- Recuperan, almacenan y borran información
- Se definen en las facetas si necesito, si añadido, si modifico y si borro de las propiedades de instancia.
- Sus características
  - Se definen en marcos clase y permanecen latentes hasta que se dispara su ejecución desde un marco instancia.
  - Cuando se dispara, el procedimiento asociado se ejecuta con los valores del marco de instancia.
  - Unos demonios pueden disparar a otros si necesitaran consultar o modificar una propiedad.

## 6.6 Ventajas de los marcos frente a las redes semánticas

1. Los marcos permiten agrupar y estructurar la info asociada a cada entidad de dominio. Las redes dispersan la info de una misma entidad.
2. Las redes son declarativas, aunque no procedimentales, y los marcos contienen una parte declarativa y otra procedimental
3. Las redes no contienen valores por omisión, los marcos sí.
4. Una BC por un sistema de marcos es fácilmente ampliable frente a una formada por una red.