

Introducció a la representació del coneixement

Jordi Duran Cals
Jordi Conesa i Caralt
Robert Clarisó Viladrosa

PID_00199490



Els textos i imatges publicats en aquesta obra estan subjectes –llevat que s'indiqui el contrari– a una llicència de Reconeixement-NoComercial-SenseObraDerivada (BY-NC-ND) v.3.0 Espanya de Creative Commons. Podeu copiar-los, distribuir-los i transmetre'ls públicament sempre que en citeu l'autor i la font (FUOC. Fundació per a la Universitat Oberta de Catalunya), no en feu un ús comercial i no en feu obra derivada. La llicència completa es pot consultar a <http://creativecommons.org/licenses/by-nc-nd/3.0/es/legalcode.ca>

Índex

Introducció	5
Objectius	6
1. El coneixement	7
1.1. El coneixement en la intel·ligència artificial	7
1.2. IA simbòlica enfront d'IA subsimbòlica	8
1.3. Classificació del coneixement	10
1.3.1. La jerarquia del coneixement	11
1.3.2. Coneixement tàcit enfront de coneixement explícit	12
1.3.3. Coneixement declaratiu enfront de coneixement procedimental	15
1.3.4. Coneixement relacional	16
1.3.5. Coneixement inferencial	17
1.3.6. Coneixement heretable	17
1.3.7. Coneixement lingüístic	22
2. La representació del coneixement	24
2.1. Esquemes de representació del coneixement	25
2.2. Propietats d'un esquema de representació	25
2.3. Etapes de la representació del coneixement	26
2.4. Reptes de la representació del coneixement	27
2.4.1. El problema de modelitzar el món	28
2.4.2. El problema del sentit comú	29
2.4.3. El problema del llenguatge natural	31
2.5. Els esquemes lògics	32
2.5.1. La implementació	34
2.6. Els esquemes basats en xarxes	35
2.6.1. Les xarxes semàntiques	36
2.6.2. Els grafs conceptuals	38
2.6.3. La implementació	39
2.7. La representació estructurada	41
2.7.1. Els marcs	41
2.7.2. Els <i>scripts</i>	44
2.8. La representació procedimental	47
2.8.1. Les regles de producció	48
2.8.2. Els programes	49
2.9. Les ontologies	50
2.10. Qüestions que s'han de tenir en compte	52
2.10.1. Les relacions	53
2.10.2. Els atributs	54

2.10.3. Restriccions d'integritat	54
2.10.4. L'elecció de la granularitat	55
2.10.5. Relacions individuals i col·lectives	55
2.10.6. El repte del coneixement incert i heurístic	56
Exercicis d'autoavaluació.....	59
Solucionari.....	60
Glossari.....	65
Bibliografia.....	67

Introducció

En el material següent es presenta una de les principals àrees de la intel·ligència artificial (IA), la representació del coneixement, que és fonamental per a la resolució d'una gran varietat de problemes. Aquesta àrea és molt àmplia i abraça tant aspectes teòrics com els més purament epistemològics, i també els problemes més pràctics de tractament de dades. Aquest material se centrarà en el vessant més aplicat: les estratègies que permeten codificar el coneixement humà per a ser utilitzat computacionalment. De fet, tot i que s'enfoquen en l'àrea d'IA, els principis de la representació del coneixement s'utilitzen en altres subàrees de la informàtica, com per exemple la modelització conceptual o les bases de dades.

Tal com indica el seu nom, la representació del coneixement té la intenció de definir mecanismes per a representar el coneixement per mitjà de símbols i facilitar-ne així el tractament. Aquests mecanismes són els encarregats de nodrir sistemes intel·ligents adreçats a resoldre tasques complexes que requereixen raonar sobre un determinat domini d'aplicació. Sovint, l'eficàcia dels sistemes intel·ligents depèn d'una bona representació de les dades, de la informació o del coneixement del domini en el qual actuen. Per tant, estem davant d'una àrea clau en el camp de l'enginyeria informàtica.

Abans de començar, cal tenir present que la tasca de modelitzar la realitat és una tasca impossible amb les tècniques i eines actuals. La realitat té massa variables, casos particulars i relacions causals perquè pugui ser descrita totalment. De fet, representar de manera eficient tan sols una part de la realitat, la rellevant per a un problema concret, ja és una feina d'unes dimensions astronòmiques. En aquest material mostrarem alguns projectes de recerca que fa dècades que persegueixen aquest objectiu tan ambiciós.

Epistemologia

L'epistemologia és la ciència que reflexiona filosòficament sobre el procés del coneixement humà i els problemes que s'hi presenten.

Objectius

Amb l'estudi d'aquest mòdul didàctic assolireu els objectius següents:

- 1.** Saber analitzar el coneixement per mitjà de la classificació de la seva procedència i forma.
- 2.** Conèixer la complexitat i els límits de representar el coneixement.
- 3.** Conèixer amb tots els detalls diferents representacions del coneixement.
- 4.** Saber exposar els conjunts d'esquemes de representació i les seves principals aplicacions.
- 5.** Saber quin formalisme específic és més apropiat per a representar cada tipus de coneixement.

1. El coneixement

El terme **coneixement** s'utilitza de forma quotidiana amb un significat molt genèric.

Coneixement

“Fets, informació i habilitats adquirides per mitjà de l'experiència o l'aprenentatge per la comprensió teòrica o pràctica d'un tema. També, consciència o familiaritat adquirida per l'experiència d'un fet o una situació” (*Oxford Dictionary*).

En aquest apartat, estudiarem el paper del coneixement en un àmbit concret de la informàtica, la intel·ligència artificial, i considerarem els diferents tipus de coneixement que hi intervenen.

1.1. El coneixement en la intel·ligència artificial

Un dels objectius de la intel·ligència artificial (IA¹) és el desenvolupament de tècniques i mètodes que permetin a un sistema informàtic resoldre problemes de manera intel·ligent, és a dir, tenint en compte el context i la informació disponibles per aconseguir l'objectiu que es busca. Resoldre aquest tipus de problemes pot requerir determinades habilitats, com la capacitat d'aprendre, de raonar o de planificar, entre d'altres.

⁽¹⁾IA és l'acrònim d'*intel·ligència artificial*.

Un requeriment implícit en el desenvolupament d'un sistema intel·ligent és la capacitat de representar i d'utilitzar el coneixement necessari per a resoldre el problema, tal com ho faria un expert humà.

Exemple 1. Conducció de vehicles terrestres: tipus de coneixement requerit

Una aplicació pràctica de la intel·ligència artificial és el desenvolupament de sistemes capaços de conduir vehicles terrestres sense la supervisió dels humans. Un sistema com aquest ha de ser capaç de tractar i d'interrelacionar coneixement de diferents àmbits:

a) Coneixement sobre el mateix vehicle:

- Les seves característiques: alçària, amplada, longitud, pes, etc.
- El seu estat: velocitat, combustible disponible, nombre d'ocupants, càrrega total, possibles avaries, etc.

b) Coneixement sobre el problema de la conducció:

- Les normes del Codi de circulació
- Les lleis físiques relacionades amb el moviment i l'acceleració, etc.

c) Coneixement sobre l'entorn:

Observació

En aquest punt, i fins que no definim d'una manera més precisa el terme *coneixement*, farem un abús del llenguatge per simplificar la lectura i utilitzarem la paraula *coneixement* quan realment ens referim a dades o a informació.

- Els altres vehicles de la via: posició, direcció i velocitat
- Altres objectes rellevants per a la conducció: carrils, senyals de trànsit, obstacles a la ruta, etc.
- Les condicions climatològiques: pluja, neu, boira, etc.
- Les condicions de la carretera: il·luminació, tipus de via, incidències, etc.

Des que el 1956 apareix el terme IA en la *Darmouth conference* fins ara, els professionals d'aquesta àrea han construït sistemes que incorporen coneixement de manera més o menys explícita. Molts sistemes intel·ligents utilitzen de manera intensiva el coneixement d'algun **domini**, el camp d'aplicació on definim el nostre problema. El ventall d'exemples és molt gran i variat, i tenim com a mostra sistemes de diagnosi mèdica i d'anàlisi lingüística. Però en molts casos el coneixement que contenen no està representat explícitament, fet que no el fa extensible ni manipulable.

Els dominis de discurs

Un domini de discurs (o simplement domini) és una àrea de coneixement que engloba el coneixement rellevant que volem representar. Exemples de coneixement poden ser el subconjunt de la física dinàmica necessària per a entendre la conducció dels vehicles, o bé la informació sobre les funcions d'un vehicle, sobre la seva interfície, etcètera.

Exemple 2. Jugar a escacs: ús de la força bruta enfront de l'ús de coneixement

Un programa informàtic que juga a escacs pot intentar trobar la millor jugada en cada posició utilitzant únicament "la força bruta": provant un gran nombre de combinacions de jugades. Però el programa serà molt més efectiu si és capaç de reconèixer i aprofitar conceptes dels escacs com ara *obertura*, *final de rei* i *peons* o *escac mat amb torre i rei*. En aquests casos, i aprofitant aquest coneixement, el programa podria ajustar el procés de cerca, o bé utilitzar una biblioteca de jugades predefinides.

En conseqüència, la comunitat científica ha reconegut la necessitat de **modelitzar el coneixement**.

Modelitzar el coneixement

"Si el propòsit de treballar amb la IA és modelitzar la ment humana o dissenyar sistemes intel·ligents, necessàriament inclou un estudi del coneixement" (Aaron Sloman, 1979).

Com veurem a continuació, dins el camp de la IA hi ha diferents concepcions sobre com utilitzar el coneixement en la resolució de problemes.

1.2. IA simbòlica enfront d'IA subsimbòlica

La disciplina de la IA es classifica en un conjunt d'àrees, com ara la resolució de problemes i cerca, la intel·ligència artificial distribuïda, la representació del coneixement i l'aprenentatge computacional². Les dues darreres àrees estan estretament lligades entre si i representen dos paradigmes diferenciats en el camp de la IA, la **IA simbòlica** i la **IA subsimbòlica**:

⁽²⁾En anglès, *machine learning*.

a) IA simbòlica

En la IA simbòlica, el coneixement es representa per mitjà d'unitats discretes (símbols), que es poden combinar seguint certes normes en un formalisme o llenguatge d'estructures més complexes: propietats, fórmules, regles, relacions, sentències...

Exemple 3. Norma del domini de circulació viària

En un sistema de conducció automàtica, el codi de circulació viària és crucial per a circular de manera segura i legal. Aquest coneixement està descrit com una llista de normes, cadascuna de les quals es pot representar de manera simbòlica. Un exemple podria ser la regla següent:

Si circulem per una via als EUA, **llavors** cal circular per la dreta.

Històricament, la IA simbòlica és l'aproximació original cap a la IA que es coneix com a *IA clàssica*. El camp de la representació del coneixement s'engloba dins d'aquest paradigma i serà l'objecte d'estudi d'aquest mòdul.

b) IA subsimbòlica

En la IA subsimbòlica, el coneixement es transmet per mitjà de propietats implícites dels objectes.

Aquest tipus de representació, utilitzada en l'àrea de l'aprenentatge computacional, apareix com a conseqüència de les limitacions de la IA clàssica: en alguns problemes, els formalismes existents són inapropiats per a capturar tota la informació requerida amb el nivell de detall necessari. Per exemple, això passa en problemes en què es treballa amb coneixement incert, relacions espacials, etc. En altres casos, el problema és la falta d'un expert del domini o estar davant d'un domini de dimensions considerables, que fa inviable representar-lo de manera simbòlica.

Exemple 4. Reconeixement facial: exemple d'IA subsimbòlica

Un problema típic en visió per computador és identificar un objecte, una persona o una ubicació a partir d'una imatge. Encara que un objecte tingui una forma molt característica, reconèixer-lo pot arribar a ser un problema molt complicat. Poden variar l'angle de visió, la distància a la qual ens trobem o el nivell d'il·luminació, o bé hi pot haver obstacles que l'ocultin parcialment. Així doncs, no és possible descriure simbòlicament totes les maneres diferents de percebre un objecte en una imatge.

En aquestes situacions, la solució passa per utilitzar tècniques d'aprenentatge basades en exemples: màquines de suport vectorial³, xarxes neuronals, etc. Aquestes tècniques analitzen un conjunt d'exemples per a calcular pesos, probabilitats o heurístiques que s'utilitzaran en el procés de decisió. És a dir, intenten extrapolar el coneixement a partir dels exemples, tot i que aquest coneixement no s'acaba explicitant sinó que queda implícit en els valors calculats.

Reflexió

En l'apartat 2 d'aquest mòdul estudiarem diferents formalismes de representació, en què explicarem, entre d'altres, les representacions lògiques, les xarxes semàntiques o els marcs (*frames*).

⁽³⁾En anglès, *support vector machines*.

Exemple 5. Xarxes neuronals: exemple d'IA subsimbòlica

Una xarxa neuronal és un model matemàtic d'inspiració biològica que imita el funcionament de les neurones del sistema nerviós. En comptes d'assignar una tasca diferenciada a cada neurona, el comportament de la xarxa està determinat per les connexions entre les neurones.

En la fase d'entrenament, el pes o la rellevància de cada connexió es configura a partir dels exemples disponibles. Un cop acabat l'entrenament, aquests pesos s'utilitzaran per a donar resposta davant nous escenaris.

D'aquesta manera, una xarxa neuronal pot "aprendre" a reconèixer formes (cares, senyals de trànsit...) a partir d'exemples. Malgrat tot, no podrà "explicar" de manera simbòlica el perquè de les seves decisions, ja que aquesta informació queda implícita en els valors dels pesos de les connexions. Això fa molt difícil manipular o ampliar aquest coneixement.

Exemple 6. Algoritmes genètics: exemple d'IA subsimbòlica

Els algorismes genètics són un esquema de cerca que també té inspiració biològica. Davant el repte de trobar la millor solució a un problema segons un cert criteri d'optimalitat, aquest mètode intenta imitar el procés d'evolució i selecció natural.

L'esquema parteix d'una població on cada individu és una solució potencial. En cada etapa de la cerca (*generació*), s'avalua la qualitat de les solucions, s'elimina una part de les solucions menys prometedores i es mantenen les millors. A més, en la nova generació s'introdueixen **mutacions** en forma de canvis a solucions existents i **encreuaments** entre solucions diferents per a generar nous individus. El procés continua durant un cert nombre d'iteracions i, finalment, se selecciona la millor solució trobada.

Tot i que aquest procés pot aconseguir solucions molt bones a problemes de cerca complexos, no permet extreure coneixement del procés. Per exemple, no permet explicar quins factors cal tenir en compte per a construir bones solucions, ja que la cerca té un component molt elevat d'aleatorietat.

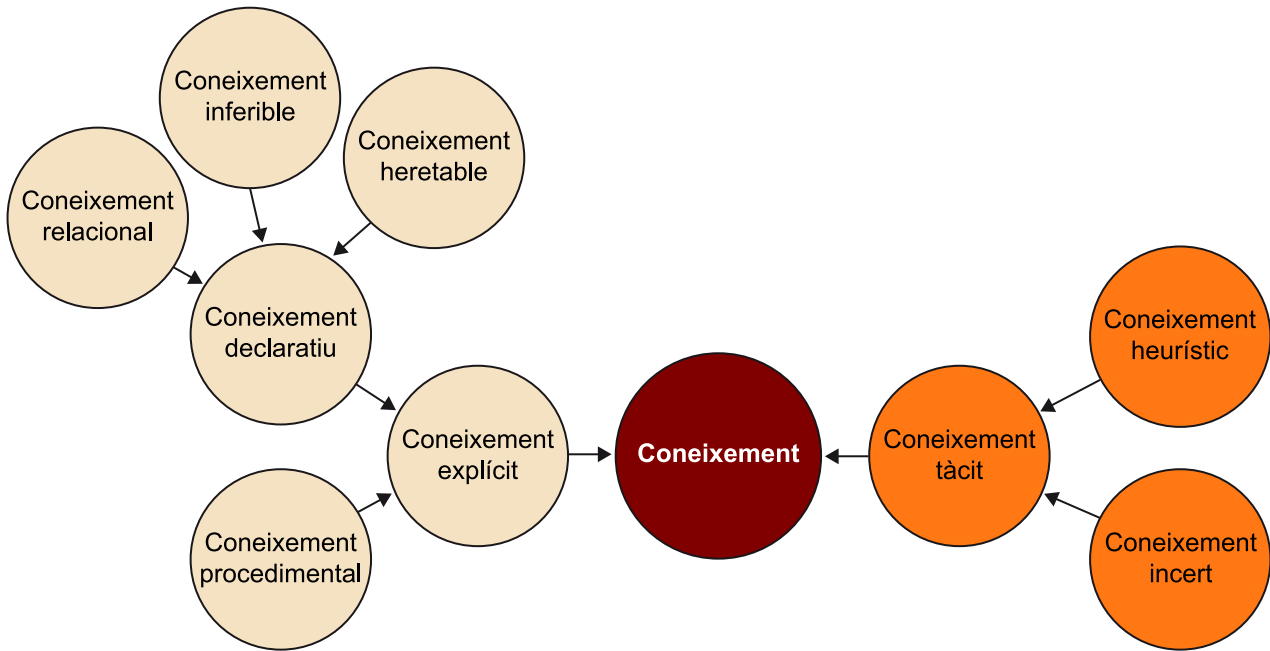
Camps d'aplicació de la IA

Els camps més comuns on s'apliquen mètodes d'IA subsimbòlica són la visió per computador, la robòtica i el processament del llenguatge natural, entre d'altres.

1.3. Classificació del coneixement

En el decurs de centenars d'anys d'estudi del coneixement, en camps com la filosofia o la religió, s'han considerat nombroses classificacions possibles. A continuació, presentem de manera gràfica una classificació dels tipus de coneixement que poden aparèixer en la modelització d'un domini. En el decurs dels subapartats següents, n'exposarem les característiques.

Figura 1. Tipus de coneixement i les seves relacions



1.3.1. La jerarquia del coneixement

Per a estudiar d'on procedeix i com s'obté el coneixement, alguns teòrics com J. Rowley (2007) utilitzen la relació jeràrquica entre els conceptes de *dades*, *informació*, *coneixement* i *saviesa* (DIKW⁴).

⁽⁴⁾De l'anglès, *data, information, knowledge and wisdom*.

La jerarquia del coneixement comença al nivell més baix a partir d'unes **dades**, de les quals obtenim **informació** en organitzar-les i analitzar-les. Seguidament, la interpretació o avaluació de la informació ens dona el **coneixement**. Finalment, la **saviesa** és la comprensió dels principis que el coneixement comprèn interiorment.

A continuació, podem veure amb més detall cadascun d'aquests nivells:

a) Dades

Les dades són valors, fets i evidències sobre un aspecte concret d'un objecte o concepte.

Exemple 7. Significat de dades

Un senyal de trànsit pot tenir forma geomètrica d'octàgon regular, triangular, rodona o quadrada. Per exemple, un conjunt de dades sobre un senyal podrien ser: un senyal de color vermell, amb la forma geomètrica d'un octàgon regular i amb el text "STOP" inscrit a l'interior.

b) Informació

La informació és una idea elaborada sobre un objecte o concepte sobre la base de la combinació de les seves dades.

Exemple 8. Significat d'informació

Un senyal de *stop* té la forma geomètrica d'un octàgon regular amb fons de color vermell i la paraula "STOP" inscrita en majúscules en color blanc. Per tant, informació seria discernir que el senyal descrit en l'exemple anterior és un senyal de *stop*.

c) Coneixement

El coneixement és informació contextual que ajuda a entendre una situació real i defineix l'experiència.

Exemple 9. Significat de coneixement

Si trobem un senyal de *stop* en el sentit de la circulació, llavors hem d'aturar el vehicle, buscar una bona visibilitat i cedir el pas, si escau, abans de continuar. Aquest coneixement es crea a partir de la informació que tenim: hi ha un senyal de *stop*, és en la direcció de la marxa i la informació que el senyal de *stop* controla la precedència de pas de diferents vehicles en un encreuament.

d) Saviesa

La saviesa és l'esglaó més alt de la comprensió, és a dir, la comprensió completa dels efectes i els resultats del coneixement.

Exemple 10. Significat de saviesa

No aturar-se en un *stop* comporta un risc molt elevat de patir un accident. Per precaució, encara que l'*stop* afecti altres vehicles, és important estar alerta i comprovar que han vist l'*stop* i s'han aturat. Aquest raonament parteix del fet que els senyals de *stop* acostumen a situar-se en encreuaments on el conductor té poca visibilitat o poc marge de reacció.

1.3.2. Coneixement tàcit enfront de coneixement explícit

Un primer criteri de classificació del coneixement és la possibilitat de comunicar-lo. Segons aquest criteri podem distingir entre el coneixement tàcit (o implícit) i el coneixement explícit:

Taula 1

Coneixement tàcit	Coneixement explícit
<ul style="list-style-type: none"> • Existeix encarnat dins de l'ésser humà. • És difícil d'expressar en una estructura. • És complicat de comunicar o compartir. • S'elabora a partir de l'experiència, els actes i també d'una visió subjectiva. 	<ul style="list-style-type: none"> • Pot existir fora dels éssers humans. • És fàcil d'estructurar i expressar. • Pot ser compartit, processat i guardat. • S'elabora a partir de conceptes, processos, procediments i principis.

Observació

Fixeu-vos que hi ha un paral·lelisme clar entre el coneixement explícit i tàcit i la IA simbòlica i subsimbòlica.

Exemple 11. Coneixement tàcit i coneixement explícit

Un exemple de coneixement tàcit és el reconeixement facial. Les persones tenim la capacitat de reconèixer la cara d'una persona coneguda entre un milió, tot i que no sapiguem descriure amb paraules com ho sabem.

Per contra, un exemple de coneixement explícit és una ruta entre dos punts: és factible donar un mapa i una llista d'indicacions per a anar des del lloc d'origen fins al de destinació.

Òbviament, en l'estudi de la representació del coneixement ens centrarem en el coneixement explícit, ja que és el que podem codificar i comunicar.

En el procés de creació del coneixement explícit estan involucrats els objectes següents:

a) **Fets:** dades o instàncies específiques i úniques.

Exemple 12. Fets

Exemples de fets podrien ser: el nombre d'accidents que hi ha hagut el 2012, que un dels precursors de la informàtica va ser Alan Turing o que el color de fons d'un senyal de *stop* és vermell.

b) **Objectes o individus:** elements concrets del món real, entenent com a elements qualsevol objecte concret (persones, animals, edificis, documents...) o abstractes (temps, hora, nombres, paraules...). Els individus poden ser tangibles o no. En alguns àmbits també es fa servir el terme *instància* per a referir-se a individus/objectes.

Observació

A partir d'aquí farem servir els termes *objecte*, *instància* i *individu* de manera indistinta per a referir-nos al mateix concepte.

Exemple 13. Instàncies

Exemples d'individus serien el nombre 3, el comandant Spock o Leonard Nimoy (què és l'actor que el representa).

c) **Conceptes o classes:** classes d'elements, paraules o idees que són coneguts per un nom comú i comparteixen característiques també comunes. Els conceptes permeten classificar objectes en funció de les seves característiques comunes, com per exemple la classe *Estudiants de l'assignatura "Representació del coneixement"*, que inclouria tots els estudiants d'aquesta assignatura. Altres exemples de classe serien la classe *Actors*, la classe *Estudiants*, la classe *Factures*, etcètera.

d) **Relacions entre classes:** indiquen possibles relacions entre les classes o entre les seves instàncies. El nombre d'elements que permet relacionar una relació acostuma a ser fix i és determinat per la seva aritmeticitat. Tot i que és possible tenir relacions d'aritmeticitat superior a dos, és difícil trobar representacions que utilitzin relacions amb una aritmeticitat de quatre o superior. Les relacions acostumen a tenir una semàntica que indica quina és la relació entre els objectes relacionats i un conjunt de restriccions d'integritat. Les restriccions d'integritat poden ser

molt variades, però per norma general permeten indicar quants objectes poden participar en la relació, de quin tipus han de ser aquests objectes i la seva obligatorietat a participar en la relació.

Exemple 14. Relacions

Exemples de relacions serien:

- La relació anomenada *estudia a* que relaciona la classe *Estudiants* amb la classe *Assignatures* i permet representar les assignatures que cursa cada estudiant.
- La relació anomenada *mare biològica* que relaciona la classe *Dona* amb la classe *Persona*. En aquest cas s'hi podria afegir una restricció d'integritat per a indicar que tot humà ha de tenir per força una mare i només una; per tant, restringint el nombre de mares que pot tenir una persona (a un) i definint la obligatorietat que tota persona participi en la relació (tota persona té mare).
- La relació *ha aconseguit medalla* entre les classes *Atleta*, *Jocs Olímpics*, *Tipus de medalla* i *Prova*, que permet representar les medalles que obtenen els atletes en unes proves d'uns jocs olímpics, i de quin tipus són les medalles obtingudes. Com es pot comprovar, aquesta relació seria quaternària, és a dir, tindria una aritat de quatre.

e) Principis: directrius, regles que s'han de satisfer, normes i paràmetres que regeixen i permeten prendre decisions i extreure conseqüències. Els tipus de principis més coneguts són les restriccions d'integritat, les regles de derivació i les regles heurístiques:

- Les restriccions d'integritat defineixen condicions que totes les instàncies han de satisfer per tal de ser vàlides. Exemples clars de restricció d'integritat són "l'edat d'una persona no pot ser negativa" o "un home no pot ser la mare biològica d'una persona".
- Les regles de derivació permeten calcular automàticament instàncies (o valors d'aquestes) a partir d'una altra informació existent. La regla que permet calcular l'edat d'una persona a partir de la data de naixement seria un exemple de regla de derivació.
- Les regles heurístiques permeten indicar condicions que es compleixen en molts casos, però que no se satisfan sempre. Exemples de regles heurístiques serien "els pianos són grans" o "els animals de companyia són afaibles".

f) Esdeveniments (processos/procediments): indiquen accions que es poden fer sobre el coneixement descrit o esdeveniments que poden succeir en el domini d'un discurs. Es poden definir de manera declarativa o procedimental. En el primer cas es definirà "quin és el resultat esperat", mentre que en el segon cas es definiran les accions i decisions necessàries per a resoldre una tasca i en quin ordre s'han d'executar, és a dir, "com arribar al resultat esperat".

1.3.3. Coneixement declaratiu enfront de coneixement procedimental

Un segon criteri per a classificar el coneixement és el tipus de pregunta a què dóna resposta: “què?” o “com?”. Segons aquest criteri, podem distingir entre coneixement declaratiu (o descriptiu) i coneixement procedimental (o operacional).

El **coneixement declaratiu** se centra en la representació dels fets o les afirmacions, és a dir, ens indica el “què és cert o fals”, referint-se als objectes i als esdeveniments.

Un símil

Fent un símil, podem veure el coneixement declaratiu com una base de dades d'alt nivell.

Des del punt de vista de la descripció del món, el coneixement declaratiu és un **coneixement factual**. És a dir, estructura els fets, els significats, els conceptes i el coneixement sobre el món extern que les persones podem adquirir. A més, es pot compartir perquè és independent de l'experiència personal i del context espacial i temporal.

Així doncs, el coneixement declaratiu es limita a enunciar informació, però sense tenir en compte els processos ni algorismes que s'hi aplicaran. Això és molt útil per a facilitar l'agregació de nou coneixement en l'estructura que el sustenta, però també ens limitarà en el procés d'inferència. Cal dir, però, que probablement és el coneixement més utilitzat i alhora el més conegut.

Exemple 15. Coneixement declaratiu en el context de conducció de vehicles

Un cotxe està ben aparcat quan està a una distància prudencial dels vehicles circumdants i de les voreres, sense tocar cap d'aquests elements, i no infringeix cap norma viària.

Observeu que el coneixement descrit és declaratiu, perquè descriu què vol dir aparcar bé, però no com fer-ho.

Els tipus de coneixement declaratiu són el relacional, l'inferible i l'heretable. Abans d'analitzar-los amb detall, estudiem primer com és el coneixement procedimental.

El **coneixement procedimental**⁵ ens indica “com es realitza una cosa”, és a dir, se centra en les accions que s'han de dur a terme per a assolir un objectiu concret.

⁽⁵⁾Enfocat a la tasca per a portar a terme un objectiu.

Podem veure el coneixement procedimental com un conjunt de passos i instruccions que especifiquen el conjunt d'accions per a dur a terme una tasca. Com a avantatge envers el coneixement declaratiu, és un tipus de coneixement molt fàcil d'aplicar, però per contra es fan més difícils la inferència, la modificació i la verificació.

Exemple 16. Coneixement procedimental

És possible saber tota la teoria per a la conducció d'un cotxe, per exemple, quin pedal és l'accelerador, quines direccions pren el canvi de marxa o què volen dir els senyals de trànsit, però aquests coneixements declaratius no ens serveixen per a saber conduir un cotxe. El coneixement de la conducció és procedimental, molt diferent de conèixer una col·lecció de fets.

També relacionat amb l'exemple anterior, el coneixement que explica com aparcar un cotxe és procedimental: comprovar que s'hi pot aparcar, situar-se en paral·lel del cotxe anterior adjacent a l'aparcament, introduir el vehicle en l'espai fent marxa enrere...

A diferència del coneixement declaratiu, el coneixement procedimental inclou també els processos per a usar i manipular el coneixement declaratiu. Els mecanismes per a representar el coneixement procedimental els tenim en els programes que coneixem tots i que podem veure com a algorismes, regles, estratègies i models.

En els apartats següents veurem amb més detall els tres tipus de coneixement declaratiu (relacional, inferencial i heretable) i el coneixement lingüístic.

1.3.4. Coneixement relacional

La manera més simple i habitual de representar fets declaratius és per mitjà d'un conjunt de relacions expressades en forma de taules, de la mateixa manera que es faria en una base de dades.

Exemple 17. Coneixement relacional

Com a exemple, mostrem la taula 2, que indica que els conductors tenen els atributs *Nom*, *Edat* i *Tipus de llicència* i permet representar instàncies d'aquests:

Taula 2. Característiques del coneixement relacional

Driver Name	Age	Licence Type
John Smith	25	B
Peter Clinton	32	A

Per si mateixa, una taula només aporta la informació (coneixement) que se'n pugui extreure directament, com per exemple saber quina llicència de conducció té una persona en concret.

D'altra banda, hi podem afegir procediments que enriqueixin el coneixement. Per exemple, hi podem afegir un motor d'inferència i així poder generar nou coneixement a partir de l'existent i validar el que ja hi ha. Així doncs, ens pot interessar saber quantes llicències de conducció hi ha d'un tipus determinat, quina és la mitjana d'edat dels conductors d'una llicència, etc. També podrem fer validacions de les dades existents, com per exemple, cap conductor de més de 75 anys pot tenir una llicència de tipus A.

Base de dades tradicionals com a mètode de representació

Els sistemes de bases de dades són per regla general molt útils per a la computació, però pel que fa a la IA són limitats. No hi ha dubte que és una forma eficient de representar i processar grans quantitats de dades. Però aleshores estem limitats a descriure només fets simples del domini i a fer raonaments senzills basats en la cerca.

1.3.5. Coneixement inferencial

El coneixement inferible o deductiu descriu el coneixement mitjançant la lògica tradicional, per tant utilitza la semàntica dels operadors i el *modus ponens* per a inferir-ne de nou.

Exemple 18. Coneixement inferencial en el cas de la conducció de vehicles

Utilitzant la lògica de primer ordre, definirem el cotxe d'en John.

1. El *Porsche d'en John* és una instància de la classe *Cotxe*:

$$\text{car}(\text{Porsche d'en John})$$

2. Tots els cotxes són vehicles:

$$\forall x:\text{car}(x) \rightarrow \text{vehicle}(x)$$

3. Tots els vehicles poden anar per terra, mar o aire:

$$\forall x:\text{vehicle}(x) \rightarrow \text{travel}(x, \text{land}) \vee \text{travel}(x, \text{water}) \vee \text{travel}(x, \text{air})$$

D'aquests enunciats per exemple podem inferir que "El Porsche d'en John pot anar per terra, mar o aire". Podem veure que com més informació tinguem d'aquests objectes més coneixement podrem inferir.

1.3.6. Coneixement heretable

La representació heretable és una estructura jeràrquica que permet representar el coneixement de forma incremental. S'obté relacionant classes amb altres classes de caràcter més general, així es permet reutilitzar el coneixement general en les noves classes i heretar totalment o parcialment el conjunt d'atributs que tenen definits. Amb aquest objectiu, s'utilitza una estructura d'arbre (taxonomia) o graf (xarxa), on els vèrtex seran classes, també anomenades *categories*, i les arestes representaran les **relacions** entre classes.

Una **categoria** és una abstracció que permet representar las característiques que tenen en comú un conjunt d'objectes. Un exemple de categoria seria la classe *Cotxe*, que permet agrupar vehicles terrestres de motor, de quatre rodes. El concepte de categoria és vital per a la representació del coneixement.

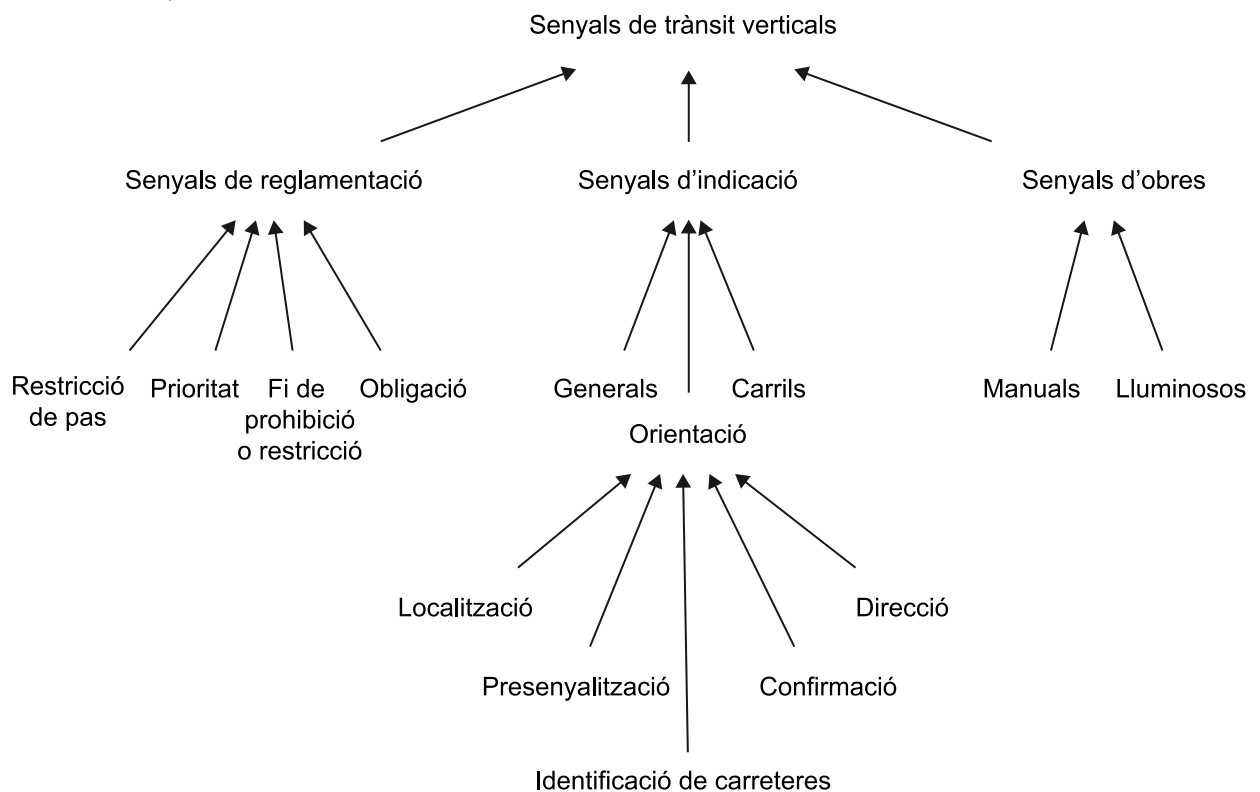
Encara que la interacció amb el món es realitza en l'àmbit d'objectes individuals, la majoria del procés de raonament té lloc en l'àmbit de les categories (Russell i Norvig, 2004).

Si organitzem les categories en subcategories per mitjà d'un arbre que les relacioni, obtindrem el que s'anomena una **taxonomia**. Convé destacar que la taxonomia és una estructura utilitzada durant segles en molts camps tècnics per a la classificació. Un dels camps és la biologia, on durant molt de temps s'ha treballat per a crear una taxonomia per a la classificació de totes les espècies vives i també extingides.

Exemple 19. Taxonomia de senyals de trànsit

En la imatge següent podem veure una classificació taxonòmica dels diferents tipus de senyals de trànsit verticals. Per exemple, el tipus de senyal *Senyals de direcció* hereta les propietats del seu node pare i aquest a la vegada del seu, i així obté la propietat de ser un senyal d'orientació i indicació.

Il·lustració 2. Exemple de taxonomia



Una manera de representar fàcilment les categories, la seva relació amb altres categories i els objectes que classifiquen és utilitzant lògica de primer ordre i la teoria de conjunts, com podem veure a continuació:

- L'operador de pertinença d'elements a un conjunt es pot utilitzar per a indicar que un objecte és un membre d'una categoria.

Exemple 20. Definició de pertinença d'un objecte a una categoria

El senyal d'indicació de *Senyal de Sortida 15 d'autopista AP7* és un objecte que pertany a la categoria de *Senyals de direcció*:

$$\text{Senyal de sortida 15 de l'autopista AP7} \in \text{Senyals de direcció}$$

- L'operador de subconjunt es pot utilitzar per a indicar que una categoria és una subcategoria d'una altra. Matemàticament *is a* correspon a la relació de pertànyer a un subconjunt (\subseteq), i *instance of*, a la relació de membre d'un conjunt (\in).

Exemple 21. Definició de subconjunt entre categories

El conjunt de *Senyals de direcció* és un subconjunt dels *Senyals d'orientació*:

$$\text{Senyals de direcció} \subseteq \text{Senyals d'orientació}$$

- La lògica de primer ordre es pot fer servir per indicar les propietats dels objectes d'una categoria.

Exemple 22. Definició de les propietats de la categoria *Senyal*

Totes les senyals estan geogràficament ubicats en algun lloc i, per tant, tenen un atribut per a indicar-ne la longitud i un altre per a indicar-ne la latitud:

$$x \in \text{Senyal} \Rightarrow \text{latitud}(x, \text{lat})$$

$$x \in \text{Senyal} \Rightarrow \text{longitud}(x, \text{lon})$$

Propietats o atributs

Una propietat (o atribut) és un cas particular de relació binària en què un dels participants és un tipus de dades. Per a obtenir-ne més informació, vegeu el subapartat 2.10.2 d'aquest mòdul.

L'herència és la forma de definir les relacions de pertinença entre categories i, per tant, permet definir que una categoria és subtipus d'una altra categoria. Conceptualment, l'herència es fa servir per a especificar categories més específiques i que, per tant, classifiquen menys objectes. Formalment, la definició de l'herència entre dues categories (que anomenarem *categoria* i *subcategoria*) defineix una restricció d'integritat de pertinença entre les dues categories, que indica que tot objecte classificat per subcategoria també ha de ser classificat per categoria. Així doncs, si definim que la categoria *Homes* és una subcategoria de *Persones*, estarem creant un esquema que garantirà que tot home (per exemple Jordi Forratge) és també una persona (Jordi Forratge està classificat també per *Persona*). Com que una subcategoria defineix de forma més concreta els elements d'una categoria, totes les relacions de la categoria són també vàlides (i per tant aplicables) a la subcategoria.

Tot i que es permet que una subcategoria hereti de més d'una categoria (cosa que s'anomena *herència múltiple*), a l'hora de la veritat no és una cosa gaire comuna en les representacions que podem trobar. No perquè no sigui útil, ni perquè no representi la realitat de manera fidel, sinó perquè molts llenguatges de representació no permeten l'herència múltiple. Si no hi ha herència múltiple, la jerarquia de categories i subcategories es pot representar en forma d'arbre. En aquest arbre, les instàncies de les subcategories es propaguen cap a la seva categoria (cap amunt en l'arbre) i les relacions de les categories es

propaguen cap a les seves subcategories (cap avall en l'arbre). Òbviament, si no hi ha herència múltiple, l'estructura ha de ser la d'un arbre, ja que per la definició de l'herència no hi poden haver cicles.

La generalització i especialització del coneixement

La manera més comuna de representar l'herència entre conceptes és via relacions de generalització i/o especialització. Una relació de generalització/especialització permet definir que hi ha una classe general, que anomenem *entitat superclasse* (també anomenada *supertipus* o *pare*), que s'especialitza en una (o més) classes, anomenades *subclasses*, *subtipus* o *filles*:

- 1) La superclasse defineix coneixement més general i acostuma a definir les característiques comunes de totes les seves subclasses.
- 2) Les subclasses defineixen informació més concreta i acostumen a definir característiques que no són comunes per a totes les subclasses de superclasse.

Formalment, podríem dir que necessitem dues relacions per a definir l'herència: una per a dir que un concepte és més general que un altre (relació de generalització), i l'altra per a dir que un concepte és més específic que un altre (relació d'especialització). La relació de generalització/especialització és bidireccional i engloba aquestes dues semàntiques. De fet, es farà servir el terme *especialització* o *generalització* en funció de si es llegeix la relació dels supertipus als subtipus o en sentit contrari.

És important també notar que l'herència, i per tant la generalització i l'especialització, també són aplicables a les relacions.

Exemple 23. Definició d'herència entre relacions

La relació *té una propietat* que relaciona les classes *Persona* i *Habitatge* i indica que una persona és propietària d'un immoble, es pot especialitzar en dues relacions més específiques: *té primera residència* i *té segona residència*.

Els termes *és un* (*is a* en anglès) i *tipus de* (*kind of* o *type of* en anglès) també se solen fer servir en alguns llenguatges de representació de coneixement per a representar relacions de generalització/especialització. Per exemple: "Canari és un Animal" o "Cadira és un tipus de Moble".

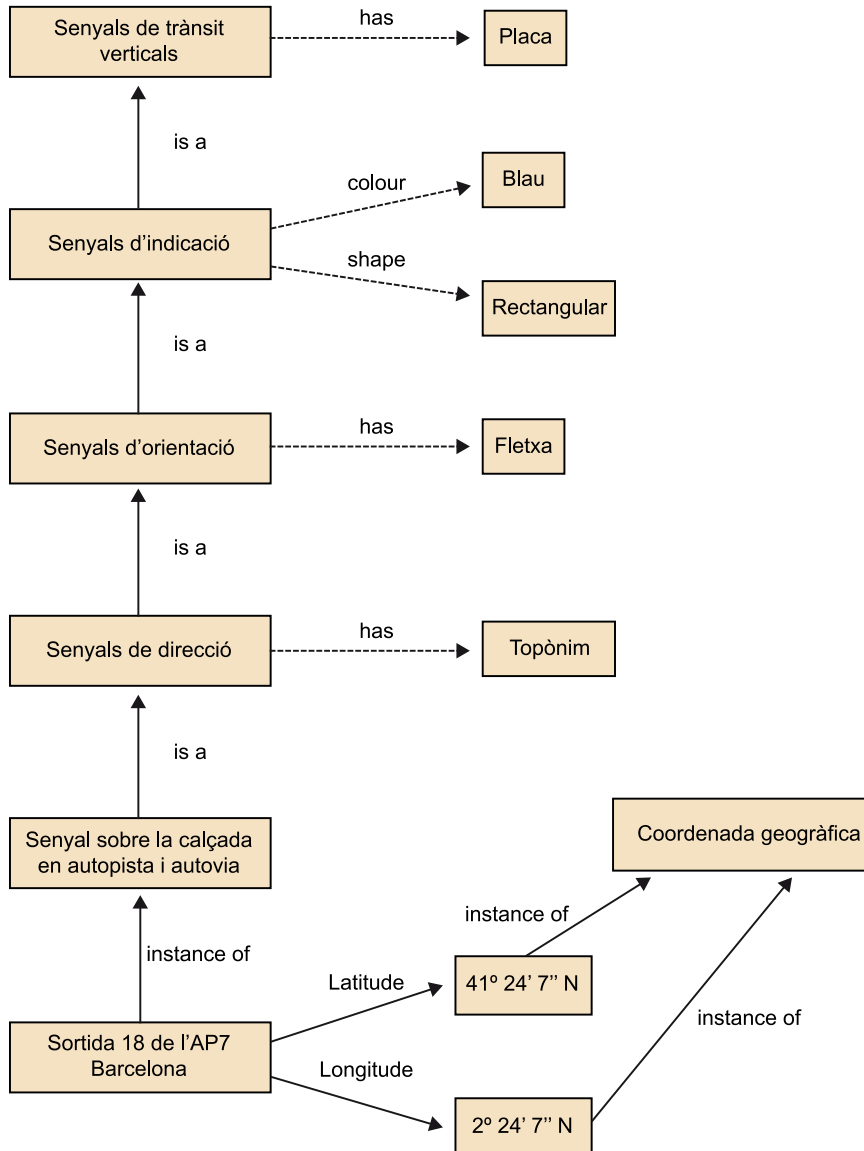
Un altre element directament afectat pel concepte d'herència són les instàncies. Per a indicar que un individu pertany a una classe es fa servir un tipus de relació anomenat *instància de* (o *instance of* en anglès). Un exemple d'instància seria "La cadira en què seus ara mateix és una instància de *Cadira*". En definir aquest coneixement, el sistema automàticament podria inferir que la teva cadira és un moble.

Exemple 24. Ús d'herència en una representació

Circulant cap a Barcelona per l'autopista AP7, en veurem indicada la sortida per mitjà d'un senyal vertical que conté el text "Sortida 18, Barcelona". Estem davant d'una instància d'un *Senyal de direcció* sobre la calçada d'una autopista. És un senyal que és fàcilment reconegut pels que coneixen el Codi de circulació, ja que és igual en forma a la resta dels senyals de la mateixa classe, és a dir, té les mateixes característiques.

De les classes a les quals pertany el senyal "Sortida 18, Barcelona" es pot inferir que aquesta és una placa rectangular i que conté una fletxa.

Il·lustració 3. Exemple d'herència en l'especialització



Aquest esquema mostra com podem representar tot aquest coneixement de manera organitzada per tal de reflectir les entitats més generals i les específiques.

A la pràctica, les fonts de coneixement declaratiu estan dividides en dos grans blocs en funció del tipus de coneixement que descriuen:

1) **Coneixement intensiu**⁶, quan es descriu el domini per mitjà de conceptes generals (classes, superclasses, relacions i principis).

⁽⁶⁾En anglès, *intensive knowledge*.

⁽⁷⁾En anglès, *extensive knowledge*.

2) **Coneixement extensiu**⁷, quan es descriu el domini mitjançant informació concreta, com per exemple utilitzant les instàncies concretes pròpies del domini.

Exemple 25. Definició de la classe *Planeta del Sistema Solar* de manera intensiva i extensiva

La classe *Planeta del Sistema Solar* es podria definir de manera intensiva o extensiva. Intensivament, la podríem definir com un subtipus de *Planeta* amb una restricció d'integritat que digui que totes les seves instàncies han d'orbitar al voltant del Sol. Extensivament, la classe es podria definir indicant que les seves instàncies són *Mercuri*, *Venus*, *La Terra*, *Mart*, *Júpiter*, *Saturn*, *Urà*, *Neptú* i *Plutó*.

1.3.7. Coneixement lingüístic

Quan parlem de coneixement lingüístic, la primera cosa que ens pot venir a la ment és un diccionari. El **diccionari** és un recull de paraules d'una llengua amb les seves definicions. Però un diccionari només conté la paraula principal, és a dir, sense derivats: plurals, flexions verbals o declinats.

Si a aquest recurs afegim relacions entre les paraules, relacions lèxiques que expressin similitud en el significat (sinonímia i antonímia), n'obtenim un **tesaurus**. Si no ens quedem aquí i fem un pas més, afegint-hi més relacions lèxiques (com la hiponímia, metonímia) i també, si agrupem les paraules per conceptes, obtindríem un recurs de coneixement declaratiu. Part d'aquest treball s'ha fet en diferents projectes, i WordNet és el més conegut.

Sinonímia, antonímia, hiponímia, metonímia

Sinonímia: Relació semàntica que s'estableix entre dues o més paraules diferents que tenen el mateix significat.

Antonímia: Relació semàntica que s'estableix entre dues o més paraules de significats oposats, és a dir, entre paraules que signifiquen el contrari o l'invers.

Hiponímia: Relació semàntica que s'estableix entre una unitat lèxica (grup de paraules amb significat propi) i una altra quan aquesta es troba inclosa dins el camp lèxic de la primera. Per exemple: *fruita* té els hipònims: *taronja*, *pera*, *poma*. La relació vista a la inversa és la **hiperonímia**.

Metonímia: Relació entre significats basada en una relació de contigüïtat. Per exemple: "Es va menjar dos plats" (es va menjar el contingut del plat).

Exemple 26. WordNet

Si consultem la paraula *car* de l'anglès en el recurs lèxic del projecte WordNet, obtindrem el conjunt de conceptes del qual forma part. Si ens fixem en el concepte principal, "vehicle de motor de quatre rodes, generalment propulsat per un motor de combustió interna", aquest concepte recull quatre sinònims o altrament dit variants de *car*: *auto*, *automobile*, *machine* i *motocar*.

A més, també podem consultar quin n'és l'hiperònim, que és el concepte: "vehicle propulsat de manera autònoma que no va per vies". Aquest té dues variants: *motor vehicle* i *automotive vehicle*. I així podríem anar navegant per la relació d'hiperonímia fins a arribar al concepte arrel: *entity* (entitat).

El projecte WordNet

El projecte WordNet és un exemple de representació utilitzada per a l'estudi del llenguatge. El projecte reuneix en una base de dades lèxica noms, verbs, adjectius i adverbis de l'anglès en conjunts de sinònims, és a dir, conceptes.

A més d'aquest coneixement que hem esmentat, altres tipus de coneixement, com el coneixement fonètic, fonològic, morfològic, sintàctic, semàntic, i fins i tot el pragmàtic, són coneixement lingüístic. Però no tot ha de ser coneixement declaratiu, ja que hi ha autors, com és el cas de Noam Chomsky, que el veuen com a procedimental. La seva posició es basa que part d'aquest coneixement és fruit d'habilitats apreses i, per tant, simulables com a procediments.

Exemple 27. Fonologia i fonètica

La **fonologia** és la ciència que estudia els fonemes, és a dir, la relació dels sons i les paraules. Molt sovint es confon amb la **fonètica**, la ciència que estudia la parla humana, en concret la producció dels sons. Totes dues disciplines tenen en comú que tracten aspectes de la parla, però des de punts diferents i complementaris.

El coneixement fonològic és representat com a coneixement declaratiu, per mitjà d'estructures d'arbre que defineixen les relacions entre els fonemes i el conjunt de síl·labes d'una llengua. Estem, doncs, davant d'un coneixement declaratiu.

En canvi, pel que fa el coneixement fonètic, aquest serà procedimental, ja que ha de representar:

- a) Processos que transformin seqüències de fonemes en articulacions del tracte vocal.
- b) Processos que converteixin vibracions vocàliques en seqüències de fonemes.

Noam Chomsky

Noam Chomsky és un dels científics més importants de la segona meitat del segle XX, estudis de la lingüística amb derivacions en les ciències de la computació.

Podeu consultar més sobre la seva figura i el seu treball a l'adreça següent:

"The Noam Chomsky Website"

2. La representació del coneixement

La **representació del coneixement** és un àmbit dins el paradigma de la IA simbòlica que estudia com especificar el coneixement en un format que doni suport en la resolució de problemes. Un altre terme molt utilitzat en l'àmbit de la representació del coneixement és el d'*enginyeria del coneixement*. L'enginyeria del coneixement és la disciplina que estudia com integrar coneixement en sistemes informàtics per tal de resoldre problemes complexos que altrament necessitarien d'intervenció humana a causa de la seva complexitat. Per tant, podem dir que l'enginyeria del coneixement no tan sols s'ocupa de la representació del coneixement, sinó de com utilitzar-lo per a solucionar un problema donat.

Els sistemes informàtics que utilitzen coneixement per a dur a terme les seves tasques s'anomenen *sistemes basats en coneixement* (*knowledge-based systems* en anglès). Pots ser el tipus de sistema basat en coneixement que exemplifica millor con s'usa el coneixement són els sistemes experts. Els sistemes experts utilitzen l'expertesa d'un especialista del domini (o d'un conjunt d'ells) prèviament formalitzada per tal de solucionar problemes complexos que requereixen una gran quantitat de coneixement, ja sigui teòric, pràctic o tàcit.

Exemple 28. Medicina i representació del coneixement

L'eficàcia d'un metge en resoldre un cas mèdic no recau solament en l'encert de la diagnosi, sinó que també és clau la seva habilitat en l'elecció del tractament més apropiat. En aquest procés l'ajudaran els seus coneixements de medicina i la seva experiència en casos anteriors.

Si es representés l'experiència del metge en una base de coneixement, es podria construir un sistema expert que emulés un metge en les tasques de fer diagnòstics, proposar tractaments i justificar tant els diagnòstics fets com els tractaments proposats.

Si fins ara hem estudiat el coneixement (la informació disponible sobre el món, el "què" se sap), ara passarem a analitzar-ne la representació (la manera com es codifica aquest coneixement, el "com" es representa el que se sap). Aquesta qüestió no és banal, ja que la manera de representar el coneixement determinarà com el podrem manipular.

Cal remarcar que serà el problema a resoldre i el domini que s'hi tracta els que ens marcaran com representar el coneixement en cada cas. A més, cal subratllar que l'eficiència de la solució que proposem estarà condicionada, a més de per la quantitat i el tipus de coneixement que es tracta, per com de bé es codifica i s'emmagatzema.

2.1. Esquemes de representació del coneixement

El principal objectiu de la representació del coneixement és facilitar l'extracció de conclusions (inferència) a partir del coneixement, i per a fer-ho és necessari expressar el coneixement en una forma computable.

Un **esquema de representació** és un instrument per a transformar el coneixement d'un domini a un **llenguatge simbòlic** dotat de **sintaxi** i **semàntica** per tal que pugui ser processat computacionalment. Cada element de la representació (símbol) codificarà un fragment d'informació pertanyent al domini en qüestió.

La **sintaxi** descriu les possibles formes de construir i combinar els elements del llenguatge. És a dir, la sintaxi especifica quin és el conjunt de sentències d'un llenguatge i quines combinacions de sentències són vàlides.

La **semàntica** determina el significat dels elements del llenguatge i la relació entre aquests i el seu referent en el domini real. És a dir, la semàntica determina de forma no ambigua què significa cadascuna de les sentències del llenguatge.

És important destacar que els esquemes de representació van més enllà de les estructures de dades o les bases de dades, les quals es preocupen exclusivament de com emmagatzemar i recuperar dades de manera eficient. Internament, la implementació d'un esquema de representació pot utilitzar estructures de dades i bases de dades, però ha d'afegir altres estructures i processos que permetin fer inferència de coneixement nou.

2.2. Propietats d'un esquema de representació

Un bon esquema de representació del coneixement possibilita un accés ràpid i acurat del contingut i en tot moment possibilita les propietats següents:

a) Representació apropiada (relatiu a la representació). Ha de tenir l'habilitat per a poder representar tot el coneixement que és necessari pel domini en qüestió. És a dir, ha de ser prou ric per tal de poder contenir les dades i la informació mínima per a donar una solució apropiada al problema.

b) Inferència apropiada (relatiu a la representació). Ha de poder manipular les estructures de la representació, de manera que en tot moment se'n puguin derivar noves estructures associades amb coneixement nou inferit de l'antic.

c) **Eficiència inferencial** (relatiu a l'ús de la representació). Ha de poder millorar el procés d'inferència mitjançant la inclusió d'heurístiques i guies que l'agilitzin, i així d'aquesta manera poder optimitzar el còmput. Per tant, ha de poder representar les característiques del problema que puguin ser explotades computacionalment.

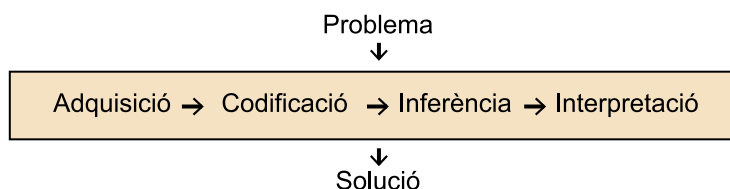
d) **Eficiència d'adquisició** (relatiu a l'ús de la representació). L'esquema ha de permetre incorporar fàcilment coneixement nou.

Altres propietats que convé tenir en compte són:

- **Claredat.** Facilitat d'identificar el coneixement representat.
- **Naturalitat.** Capacitat de representar el coneixement en la forma original, de la manera més "natural" possible, és a dir, el coneixement no s'ha de transformar. La finalitat és que petits canvis en el problema requereixin només petits canvis en l'esquema.
- **Modularitat.** Possibilitat de fragmentar el coneixement sense perdre eficiència ni eficàcia.

2.3. Etapes de la representació del coneixement

Donat un problema complex que requereix l'ús intensiu de coneixement, podem considerar les etapes següents en la seva resolució:



- El punt de partida és el **problema** que es vol resoldre, que marca quin serà el domini d'interès.
- La següent fase és l'**adquisició** de coneixement sobre el domini d'interès i el problema que es vol resoldre. Obtenir el coneixement d'un domini és una tasca molt complexa, que involucra capacitats de processat de la informació, com ara percepció, comunicació, associació i raonament. És vital obtenir tot el coneixement rellevant pel problema i evitar coneixement superflu. Així evitarem problemes (errors, males interpretacions, redundàncies, informació sobrer, dispersió...), minimitzarem els errors i millorarem el rendiment en l'accés a la informació.
- Un cop obtingut el coneixement del domini, el pas següent és la seva **codificació** en un esquema apropiat. El tipus de codificació escollit i la ma-

Reflexió

En aquest mòdul, deixarem de banda la fase d'adquisició per la seva complexitat i ens centrarem en la representació i utilització del coneixement.

nera com codificar la informació condicionaran fins a quin punt el coneixement pot ser integrat amb altres fonts d'informació i pot ser compartit amb tercers.

- Aleshores, s'aplica un motor d'**inferència** sobre el coneixement guardat en l'esquema de representació per tal d'extreure conclusions sobre el problema.
- Per tal de comunicar el resultat, el darrer pas és la **interpretació** de les conclusions obtingudes en termes de fets del domini. Així obtenim una **solució** al problema plantejat inicialment.

Exemple 29. Etapes de la representació del coneixement en la creació d'un sistema expert mèdic

En cas que vulguem crear un sistema expert per a diagnosticar malalties, una possible distribució de les tasques per fer en cada etapa és:

- 1) **Definició del problema:** donar un diagnòstic a un pacient a partir d'un conjunt de símptomes i saber-lo justificar. Es pot limitar el sistema a una malaltia o un conjunt de malalties.
- 2) **Adquisició del coneixement:** recollir informació sobre la relació entre símptomes i malalties. Fer un conjunt d'entrevistes a experts en el domini (metges) per a esbrinar com pronostiquen malalties.
- 3) **Codificació:** es codificarà el coneixement obtingut en els passos anteriors via regles causals del tipus "Si es donen el símptoma X, el símptoma Y, ..., i el símptoma Z, llavors és probable que el pacient tingui la malaltia M".
- 4) **Inferència:** quan un pacient informi el sistema dels seus símptomes, el sistema expert farà servir un motor d'inferència per a esbrinar quines de les possibles malalties és més probable que tingui l'usuari i el perquè (en cas que estigui malalt, és clar). En cas que el sistema necessiti més informació per a arribar a alguna conclusió, la demanarà al pacient.
- 5) **Interpretació:** el sistema recollirà la informació inferida en l'apartat anterior i la presentarà en un format que sigui entenedor per al pacient. En aquest cas, el sistema prepararia una resposta en llenguatge natural per explicar al pacient la seva potencial malaltia i un text breu que justifiqui aquest diagnòstic.

2.4. Reptes de la representació del coneixement

En les etapes que acabem d'identificar, apareixen diverses qüestions interrelacionades sobre les quals cal prendre decisions:

- Com s'escull quin és el domini rellevant al problema? Com delimitar quin coneixement queda dins i quin queda fora?
- Un cop triat el domini d'interès, quant coneixement s'ha d'emmagatzemar? De quin tipus?
- Com s'adquireix aquest coneixement del domini?
- Quin ús es farà d'aquest coneixement? És a dir, quin tipus d'inferències s'han de realitzar per a resoldre el problema?

- A partir de tots els punts anteriors, quin és l'esquema de representació més adequat per al problema?

Totes aquestes qüestions ens serveixen per a adonar-nos de la complicada tasca que representa escollir una bona representació. Hem de fer èmfasi que quan parlem de coneixement, a banda de representar-lo, no ens hem de descuidar en cap moment que també l'haurem de processar.

A banda d'aquestes decisions, que són particulars per a cada problema, hi ha una sèrie de reptes més generals que apareixeran en qualsevol problema de representació del coneixement que considerem: la impossibilitat de modelitzar-ho tot, el coneixement de sentit comú i el llenguatge natural. Abans de continuar, estudiarem aquests reptes en més detall.

2.4.1. El problema de modelitzar el món

L'èxit amb què un problema pugui ser solucionat dependrà directament de la nostra habilitat a poder representar el coneixement sobre el món relacionat amb el problema. Tot i així, no hem d'oblidar que no treballarem sobre el món sinó sobre una simplificació d'aquest, que inevitablement serà imperfecta o incompleta.

Per descomptat, modelitzar el món sencer és una tasca de dimensions astronòmiques. Un volum d'informació tan gran seria difícil d'emmagatzemar, i molt ineficient el seu tractament computacional.

Per aquest motiu, generalment dissenyarem estructures i procediments per tal de modelitzar-ne una part, un domini concret. De tota manera, malgrat que és un objectiu terriblement ambiciós, hi ha una varietat de projectes que pretenen registrar gran quantitat de coneixement del món en tots els seus àmbits. D'entre aquests, destaquem els projectes Cyc, ConceptNet, WordNet (explicat prèviament), The Knowledge Graph i Probase.

Cyc

Cyc és un projecte privat iniciat el 1984 en el si d'una empresa dels EUA. Des d'aleshores, el projecte ha reunit milions de fets, conceptes, relacions entre els quals, idees, regles, heurístics, etcètera. Alguns exemples del seu contingut són "Cada gos és un mamífer", "Els gossos moren" o "Els animals de companyia tendeixen a ser afables". Tot aquest coneixement s'ha codificat mitjançant un formalisme, el llenguatge anomenat *Cycl* (basat en el llenguatge Lisp). Cal destacar la publicació en obert d'una part del projecte sota el nom d'*OpenCyc*.

El projecte Cyc

Trobareu més informació sobre el projecte Cyc a la pàgina web d'[OpenCyc](#).

Exemple 30. Definició d'informació en Cyc

La regla “Els gossos són mamífers” es defineix en CycL com:

```
(#$gensl #Dogs #Mammals)
```

en què *Dogs* és la classe que representa els gossos, *Mammals* la classe que representa els mamífers i *gensl* és la relació de generalització/especialització de CycL.

ConceptNet

ConceptNet és un projecte que neix dins del Massachusetts Institute of Technology (MIT), en el MIT Media Lab, amb la col·laboració de nombrosos organismes i empreses d'arreu del món. L'objectiu és descriure el coneixement bàsic, cultural i científic del món que les computadores han de saber. El fruit d'aquest treball és un immens graf on els vèrtexs representen conceptes i les arestes les propietats que els relacionen. Actualment es troba en la seva versió 5 i es distribueix sota la llicència Creative Commons.

El projecte ConceptNet

Trobareu més informació sobre el projecte ConceptNet a la pàgina web de Conceptnet 5.

Exemple 31. Definició d'informació en ConceptNet

La regla bàsica “All dogs are mammals” es defineix a ConceptNet com:

```
dog -- (belongs to the collection of) --> mammals
```

The Knowledge Graph

The Knowledge Graph és un projecte de Google que té com a objectiu crear una base de coneixement per a millorar els resultats del seu motor de cerca. Aquest coneixement representarà el món a través de les entitats que el formen: persones, llocs, pel·lícules, etcètera, i també les relacions entre aquestes. Amb aquest projecte, Google vol fer un pas endavant tecnològicament introduint coneixement semàntic dins la cerca. Aquest coneixement semàntic permet, entre altres coses, tenir informació de context dels termes cercats pels usuaris i utilitzar aquesta informació per a desambiguar cerques, refinar-les o ampliar-les.

El projecte The Knowledge Graph

Trobareu més informació sobre el projecte a la pàgina web: The Knowledge Graph; i també al blog d'aquest projecte.

Probase

Probase és l'homònim de The Knowledge Graph per a Microsoft. Probase és una base de coneixement que conté coneixement general i de sentit comú, i que té com a objectiu donar suport als programes informàtics a entendre millor la comunicació humana. La seva base de coneixement, que conté prop de tres milions de registres, es pobla automàticament a partir de milions de pàgines web que hi ha al Web i dels *logs* de cerca dels internautes.

El projecte Probase

Trobareu més informació sobre el projecte a la pàgina web: Probase.

2.4.2. El problema del sentit comú

Un acudit popular explica que el sentit comú és “el menys comú dels sentits”. Fins i tot als éssers humans ens costa aplicar el sentit comú en totes les situacions i podem arribar a cometre errades evidents.

Si volem evitar que un sistema intel·ligent cometi aquest tipus d'errades, hem de codificar el coneixement de sentit comú relatiu al problema.

Vegem com a exemple un text de Marvin Minsky en què parla sobre el sentit comú i el raonament:

“Només en matemàtiques podem dir (sentències com) «si a i b són enters, llavors $a + b = b + a$ ». Considerem ara un fet com «els ocells poden volar». Si pensem que el raonament del sentit comú és el mateix que el raonament lògic, llavors un creu que hi ha principis generals d'aquest tipus, «si Joe és un ocell i els ocells poden volar, llavors Joe pot volar». Considerem que en Joe és un estruç o un pingüí? Bé, si és així podem crear l'axioma «si Joe és un ocell i no és un estruç ni un pingüí, llavors en Joe pot volar». Però, i si en Joe està mort? O té els peus enclavats?”

Hi ha dos factors que haurem de tenir en compte a l'hora d'intentar codificar el coneixement del sentit comú:

El raonament

“El raonament és la capacitat dels humans de donar sentit a les coses per a establir i verificar fets, i també per a canviar o justificar pràctiques i/o creences.”

Nikolas Kompridis, *So we need something else for reason to mean* (2000, pàg. 271-295) [Traducció lliure]

- Hi ha una gran quantitat de coneixement de sentit comú i no és factible representar-lo tot. Caldrà triar-ne el conjunt mínim suficient per a resoldre el problema.
- Pot ser difícil identificar el coneixement de sentit comú, ja que és tan evident que acostuma a donar-se per assumit. Per tant, hi ha un risc elevat de passar per alt aquest coneixement en l'esquema de representació. Caldrà fer un esforç addicional per explicitar aquest coneixement durant la fase d'adquisició.

Exemple 32. Ús del sentit comú en problemes de posicionament

Els problemes de posicionament (*layout*) consisteixen a escollir la ubicació d'un conjunt d'objectes (portes lògiques en circuit, maquinària en una fàbrica...) en una superfície bidimensional fitada. El posicionament ha d'optimitzar criteris com ara la mida o el rendiment del sistema global.

Hi ha algunes restriccions de sentit comú a l'hora de posicionar els objectes, com ara que dos objectes no es poden superposar. Si ens oblidem de representar aquest coneixement, podem obtenir solucions errònies al nostre problema. Per exemple, podríem obtenir com a solució òptima “ubicar tots els objectes en la mateixa posició”. Aquesta solució clarament minimitza la superfície ocupada i el temps de comunicació entre objectes, però no es pot dur a la pràctica, ja que les lleis de la física impedeixen que dos objectes ocupin el mateix espai simultàniament.

2.4.3. El problema del llenguatge natural

El llenguatge que utilitzem les persones com a mecanisme per a comunicar-nos s'anomena **llenguatge natural**. Aquest nom s'utilitza en contraposició als coneguts com a **llenguatges formals**, com ara els llenguatges de programació.

L'avantatge del llenguatge natural és la seva gran **expressivitat**: permet expressar qualsevol cosa que es pugui representar simbòlicament i, per tant, pot descriure fins i tot elements d'art, fotografies o emocions. Un segon avantatge és la **gran quantitat de coneixement** que està representat en llenguatge natural.

En conseqüència, gran part dels primers estudis realitzats en el camp de la representació del coneixement estaven vinculats al llenguatge i a la cerca de la informació en el camp del llenguatge. La seva base eren les investigacions realitzades durant dècades en l'anàlisi filosòfica del llenguatge.

Malauradament, la potència del llenguatge natural genera un gran inconvenient: l'**ambigüitat**. Una frase o paraula és ambigua quan admet més d'una interpretació possible. En aquests casos, per a triar la interpretació correcta cal basar-se en el context de la frase, que normalment queda implícit.

Exemple 33. Ambigüitat del llenguatge natural

La frase següent:

“En John va veure en David al bosc amb els binocles”

té dues interpretacions possibles: la primera, que en John a través dels binocles va veure en David. Però en tenim una segona de possible: en John va veure en David, que portava uns binocles. Per tant, ens falta conèixer el context per a saber a qui pertanyen els binocles.

L'ambigüitat és el motiu pel qual els esquemes de representació es basen en llenguatges formals i no en el llenguatge natural. L'ús del llenguatge natural se centrarà majoritàriament en la interacció persona-ordinador, o bé en la fase d'adquisició del coneixement.

El **processament del llenguatge natural** (NLP⁸) és l'àmbit de la IA dedicat a la recerca sobre la comprensió i el tractament del llenguatge natural. Principalment, les tècniques de processament de llenguatge natural pertanyen a les disciplines d'aprenentatge automàtic i de mineria de dades⁹ i contribueixen al desenvolupament de mètodes de resolució de tasques com sumarització automàtica, cerca de respostes¹⁰, traducció automàtica, identificació d'entitats¹¹, desambiguació lingüística...

Llenguatge natural i llenguatges formals

Llenguatge natural: llenguatge produït espontàniament pels éssers humans, les regles del qual deriven de l'ús. S'utilitza en la comunicació humana.

Llenguatges formals: llenguatges artificials definits a partir d'un conjunt de convencions i regles que permeten una comunicació precisa.

⁽⁸⁾De l'anglès, *natural language processing*.

⁽⁹⁾En anglès, *data mining*.

⁽¹⁰⁾En anglès, *question answering*.

⁽¹¹⁾En anglès, *named-entity recognition*.

2.5. Els esquemes lògics

La lògica matemàtica defineix molts formalismes diferents per a codificar fets i propietats. Cada formalisme ofereix una certa **capacitat expressiva** per a representar el coneixement en expressions anomenades **fórmules**. A més, cada formalisme estableix unes **regles de càlcul o inferència** que permeten combinar fórmules per a extreure conclusions a partir del coneixement existent.

El procés d'inferència té un cost computacional que pot arribar a ser molt elevat i fer que no sigui factible raonar sobre una base de coneixement. Aquest cost està directament lligat a dos factors:

- La mida de la base de coneixement: com més coneixement guardem, més costós és raonar-hi.
- L'expressivitat del formalisme lògic: com més expressiu és el formalisme, les regles de càlcul són més flexibles i, per tant, hi ha moltes més formes d'aplicar-les.

L'elecció del formalisme lògic apropiat per un problema necessàriament és un compromís. Cal cercar un equilibri entre l'expressivitat (per a poder representar el coneixement necessari) i l'eficiència del procés d'inferència (per a poder raonar-hi ràpidament).

En l'àmbit de la representació del coneixement, un formalisme lògic que s'utilitza freqüentment és la **lògica de primer ordre (LPO)**, ja que permet raonar tant sobre objectes individuals com sobre conjunts d'objectes.

Un dels components fonamentals de les fórmules en LPO són els termes. Un **terme** pot ser:

- Una **variable**, (x, y, z, \dots), que pot prendre diferents valors.
- Una **funció** d'un o més arguments ($\text{suma}(x,7), \text{pare}(z), \text{potència}(x,2,y), \dots$), en què cada argument pot ser una constant, variable o una altra funció.
- Una **constant**¹² ($\text{peter}, 2, \text{blue} \dots$), que pot fer referència a un objecte del domini o al valor d'un atribut d'un objecte.

Els termes es poden relacionar entre si i definir **predicats**. Un **predicat** està format per un símbol (que indica el tipus de relació entre els termes) seguit de 0 o més termes. Per exemple, el predicat $\text{Loves}(\text{john}, \text{mother}(x))$ estableix

Lectura complementària

Per a començar la indagació en el camp del processament del llenguatge natural, un bon llibre de referència és:

C. D. Manning; H. Schütze (1999). *Foundations of statistical natural language processing*. Cambridge: MIT Press.

Indecidibles

Hi ha formalismes lògics en què el procés de càlcul pot ser tan costós que no hi ha garantia que s'acabi: el procés es podria quedar calculant indefinidament sense arribar a donar cap resposta!

⁽¹²⁾Les constants es poden veure com un cas particular de funció amb 0 arguments.

la relació *Loves* entre els termes *john* i *mother(x)*. Aquest predicat, doncs, ens podria servir per a codificar que “En John estima la mare de *x*” (on no sabem qui és *x*).

La **interpretació** que fem d’un predicat, és a dir, el significat que donem als símbols de predicat, de funció, les constants, etc. i la correspondència entre aquests i el món real ens dirà si el predicat és **cert (1)** o és **fals (0)**.

Els predicats es combinen en **fórmules** mitjançant **connectives** lògiques, com ara la conjunció (\wedge , també anomenada *and* o *i-lògica*), la disjunció (\vee , també anomenada *or* o *o-lògica*), la implicació (\rightarrow) i la negació (\neg). Un exemple podria ser: “En John estima la Lucy però no la germana de la Gemma”:

$$\text{Loves}(\text{john}, \text{lucy}) \wedge \neg \text{Loves}(\text{john}, \text{sister}(\text{gemma}))$$

La interpretació d’una fórmula es calcula a partir de la interpretació de cadascuna de les seves subexpressions, com es pot veure en la taula següent:

Taula 3. Taula de veritat de connectives lògiques: conjunció, disjunció, implicació i negació

A	B	A \wedge B	A \vee B	A \rightarrow B	\negA	\negB
0	0	0	0	1	1	1
0	1	0	1	1	1	0
1	0	0	1	0	0	1
1	1	1	1	1	0	0

Finalment, a les fórmules també es poden afegir **quantificadors**, que especifiquen el valor que prenen les variables que hi apareixen. Hi ha dos tipus de quantificadors:

- **Existencial** (\exists), que indica que la fórmula és certa per a algun valor de la variable. Per exemple, per a indicar que “En John és estimat per algú”, utilitzaríem la fórmula següent:

$$\exists x: \text{Loves}(x, \text{john})$$

- **Universal** (\forall), que indica que la fórmula és certa per a qualsevol valor de la variable. Per dir que “En John estima tothom que és estimat per en Peter”:

$$\forall x: \text{Loves}(\text{peter}, x) \rightarrow \text{Loves}(\text{john}, x)$$

2.5.1. La implementació

Prolog és un dels llenguatges més utilitzats en la representació d'esquemes lògics. Des de la seva creació el 1972 per A. Colmerauer i P. Rousell, aquest llenguatge ha contribuït substancialment a l'evolució de la IA.

Un esquema Prolog consta de dos elements: **fets** (predicats que són certs) i **regles** (patrons del tipus "si es compleixen un conjunt de condicions, es pot afirmar aquest predicat"). L'esquema respondrà a qüestions plantejades en forma d'un o més **objectius**, predicats on algun dels seus arguments poden ser variables. Prolog utilitzarà els fets i les regles per a trobar una assignació possible de valors a les variables que fan cert el predicat (resposta "Yes") o conclou que el predicat és fals per qualsevol valor de les variables (resposta "No"). Internament, s'utilitza un algorisme de *backtracking* (tornada enrere) per a considerar totes les possibles combinacions de valors de les variables.

Convencions de Prolog

En Prolog qualsevol cadena que comença per majúscula es considera una variable: un símbol que pot prendre qualsevol valor. Els noms dels predicats i funcions han de començar necessàriament per minúscula. Fixeu-vos que la convenció és diferent de l'habitual en lògica de primer ordre, on els predicats s'escriuen en majúscula i les funcions, constants i variables en minúscula.

Exemple 34. Definició de fets utilitzant Prolog

Considerem el domini de les relacions familiars. Primer definirem un conjunt de fets.

```
gender(sarah, female).
gender(peter, male).
gender(john, male).
parent(sarah, john).
parent(sarah, peter).
```

La part dreta d'una regla indica les condicions que s'han de complir i és una llista de predicats separada per comes. A la part esquerra hi ha un únic predicat, el predicat que serà cert si es compleixen totes les condicions de la part dreta.

Exemple 35. Definició de predicats en Prolog i resolució de consultes

A continuació definim un conjunt de predicats per especificar què és una persona, què és una mare i què és un germà:

- si X és el progenitor d'algú, és una persona:

```
person(X) :- parent(X, _).
```

- si X és el progenitor d'algú i és una dona, és la seva mare:

```
mother(X,Y) :- parent(X,Y), gender(X, female).
```

- X i Y són germans si tenen un progenitor Z comú:

```
sibling(X,Y) :- parent(Z,X), parent(Z,Y).
```

Aleshores, el sistema seria capaç de respondre a les qüestions següents:

```
person(sarah)           Resposta: Yes
mother(X, peter)       Resposta: Yes (X = sarah)
sibling(peter, sarah)   Resposta: No
```

Una característica molt particular de Prolog és que els valors poden ser llistes d'elements: seqüències d'elements entre claus i separades per comes, com ara [] (llista buida) , [2] , [3, 17, 1, 3, 45]. El tractament de llistes en Prolog

s'acostuma a fer recursivament: tractant el cas de la llista buida per una banda, el primer element de la llista posteriorment, i es continua tractant la resta d'elements de forma recursiva.

Exemple 36. Tractament de llistes en Prolog

Si definíssim el predicat següent per indicar si algun dels elements de la llista del segon paràmetre és un ancestre del valor passat en el primer paràmetre, "Generar una llista ordenada dels ancestres femenins de X":

```
ancestors(X, []) :- \+ mother(X, _).           (X has no mother)
ancestors(X, [ First | Rest ]) :-
    mother(First, X),           (Add X's mother as the first element)
    ancestors(First, Rest).     (Repeat recursively)
```

Aleshores, el sistema podria respondre a les preguntes següents:

```
ancestors( sarah, X ).           Resposta: Yes ( X = [] )
ancestors( john, Y ).           Resposta: Yes ( Y = [sarah] )
ancestors( peter, [Y, Z] ).     Resposta: No
```

2.6. Els esquemes basats en xarxes

A diferència de la representació lògica, que es destina a la recerca del raonament, els esquemes basats en xarxes estan principalment destinats a formalitzar el coneixement que les persones usen en el seu món. Per aquesta raó, aquest tipus de representació apareix de la mà de lingüistes i psicòlegs en treballs que tenen com a finalitat la comprensió del llenguatge natural.

Una altra apreciació respecte a la representació lògica, és que aquesta pretén assignar *valors veraders* (valor que indica en quina mesura una declaració és veritat) a les expressions lògiques basades en la interpretació del món. Aquest fet provoca que la representació lògica no sempre és suficient per a respondre qüestions com ara: "De què està fet el cautxú?".

Pel que fa a la seva forma, la representació en xarxa sosté el coneixement per mitjà d'una estructura de **graf**, en què els **vèrtexs** representen objectes o conceptes del domini del problema (per exemple el "cotxe" o el color "vermell") i els **arcs** representen normalment relacions o associacions entre aquests (per exemple "el cotxe és de color vermell"). Tant els arcs com els vèrtexs poden estar **anotats** amb etiquetes que aporten informació addicional.

Recordem

Recordem que un graf etiquetat és un graf que té associada informació als seus arcs (o arestes) mitjançant una etiqueta que conté qualsevol tipus de dades o informació útil.

Els exemples més coneguts d'esquemes basats en xarxes són les xarxes semàntiques, els grafs conceptuals i les ontologies.

2.6.1. Les xarxes semàntiques

Per a entendre què són les xarxes semàntiques¹³, cal parlar primer de lingüística. L'anàlisi lingüística està dividida majoritàriament en cinc camps d'estudi:

⁽¹³⁾Les xarxes semàntiques constitueixen una representació idònia per a la visualització del coneixement.

- El fonològic, en què s'estudia com s'assigna el so a les paraules.
- El lexicològic, en què s'analitza com es categoritzen les paraules segons la morfologia.
- El sintàctic, en què s'estudia com es valida l'ordre i estructura de les paraules en una frase.
- El semàntic, centrat en com s'assigna un significat a les paraules.
- El pragmàtic, en què s'estudia com s'extreu la intenció d'ús de les paraules en un context concret.

En aquest àmbit, les xarxes semàntiques volen proveir representacions de coneixement per a l'estudi de la semàntica. La **semàntica** intenta descriure el significat de les paraules (permet desambiguar-ne el sentit, en cas d'ambigüitat) i les condicions en què els significats poden interactuar per ser compatibles amb els altres aspectes del llenguatge.

El significat d'un concepte es defineix a partir de les relacions cap a altres conceptes. D'aquesta manera, som capaços d'inferir coneixement per mitjà de relacions apropiades com ara *is a* i *instance of*.

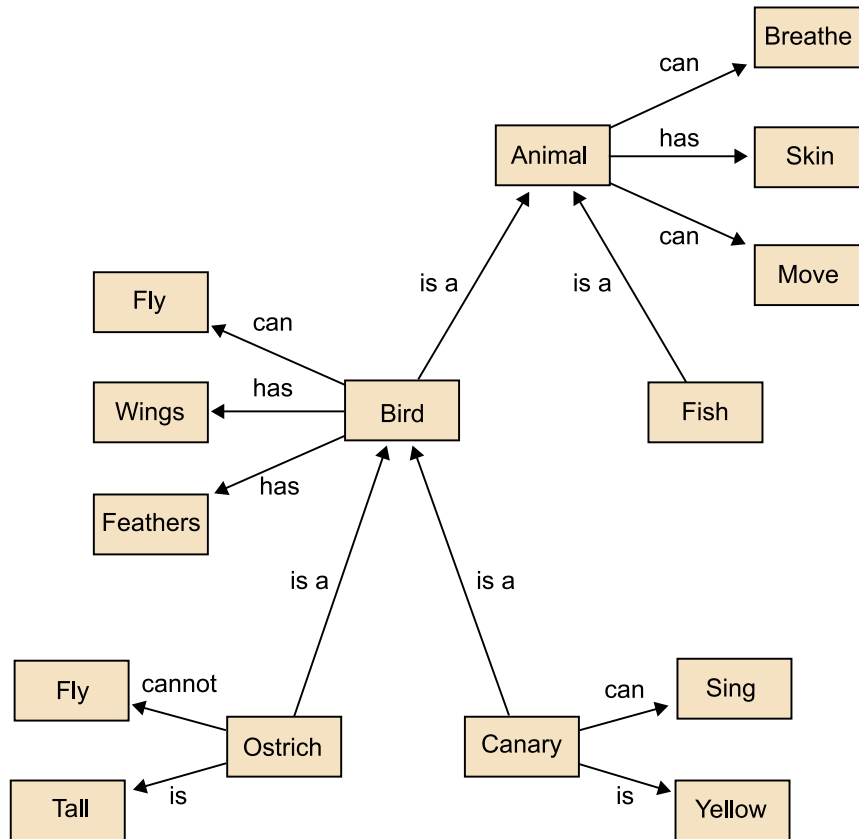
Per tant, el coneixement que mantenen les xarxes semàntiques són definicions de conceptes a més de les relacions entre elles. Aparentment, aquest és un patró d'estructura molt simple, però ha estat utilitzat amb èxit per a la construcció de moltes xarxes complexes, com és el cas del projecte WordNet, i la versió europea corresponent, l'EuroWordnet.

Moltes d'aquestes xarxes han estat proposades com a models de representació de mapes conceptuals i mentals. Per contra, d'altres s'han proposat com a components per a la comprensió del llenguatge en sistemes de raonament.

Exemple 37. Xarxa semàntica amb informació sobre ocells

El gràfic que acompanya aquest exemple (il·lustració 4) és una mostra extreta de la xarxa semàntica desenvolupada per A. Collins i R. Quillian (1969) en la seva recerca sobre l'emmagatzemament d'informació humana. És un bon exemple per a exemplificar com la jerarquia de l'estructura té la capacitat de donar respostes. És a dir, podem trobar la resposta a les preguntes següents: "Pot un canari cantar?", "És un canari un ocell?", "Pot un canari volar?" o "Respira un canari?".

Il·lustració 4. Xarxa semàntica



Estudis fets sobre com els humans emmagatzemen informació revelen que utilitzem nivells d'abstracció, on cadascun conté les propietats que esperem que siguin generalitzades. Així doncs, els éssers humans fem servir de forma interna coneixement heretable. Davant aquest coneixement, la xarxa semàntica és un mecanisme idoni per a la seva representació.

Vegeu també

El mecanisme de generalització i especialització s'explica amb més detall en el subapartat 1.3.6 d'aquest mòdul didàctic.

Exemple 38. Nivells d'abstracció

La il·lustració 4, a més, ens exemplifica un estructura amb la repartició de la informació en diferents nivells d'abstracció. En comptes d'emmagatzemar individualment totes les característiques al "canari", tenim que la característica de poder "respirar" i "moure" està al nivell d'"animal". Així també podem incloure el coneixement del "peix" en l'estructura, ja que és una especificació d'"animal".

2.6.2. Els grafs conceptuals

Estructuralment, un graf conceptual és un graf finit i connex que té la propietat de ser bipartit. Els vèrtexs del graf poden ser *conceptes* o *relacions conceptuals*. Aquest tipus de representació no associa etiquetes als arcs, sinó que són els vèrtexs de relacions conceptuals que representen les relacions entre els conceptes. Així, els vèrtexs de conceptes només tindran arcs cap a vèrtexs de relacions conceptuals, i també a la inversa. D'aquesta manera es garanteix la propietat bipartida del graf.

Els vèrtexs **concepte** representen objectes concrets o abstractes del món caracteritzats per la nostra capacitat per a formar-ne una imatge en les nostres mentes. Poden incloure tant conceptes generals, per exemple avió, telèfon o hotel, com instàncies (objectes més específics), l'avió Boeing 747 amb número de sèrie XW63783736E o l'hotel Ritz de París.

Els vèrtexs **relació conceptual** indiquen una relació que involucra un o més conceptes. L'avantatge de no utilitzar etiquetes en els vèrtexs és que això simplifica la representació de relacions amb qualsevol **aritat** (nombre de participants en una relació). Així doncs, una relació n -ària està representada per un node conceptual connectat amb n vèrtexs.

Un sistema basat en aquest tipus de representació estarà format per un conjunt de grafs, on cadascun representa una única **proposició**. Les proposicions representen part del coneixement del domini.

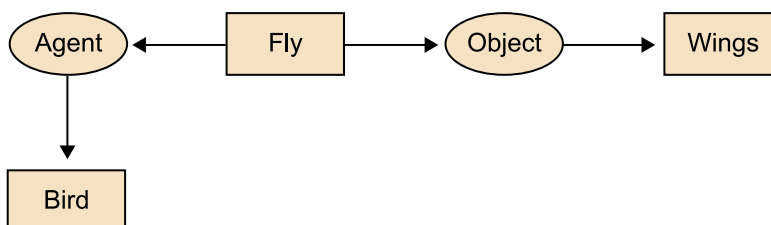
Exemple 39. Graf conceptual

La il·lustració 5 representa un graf conceptual senzill de la proposició “un ocell té plomes”. En canvi, la il·lustració 6 i la il·lustració 7, afegeixen una mica més de complexitat. En concret, el graf conceptual 2 representa “l'ocell pot volar amb les ales”. Fixeu-vos que “volar” es relaciona conceptualment amb “ocell”, ja que és l'agent de l'acció. També es relaciona amb “ales”, en què aquestes són l'objecte de vol. D'altra banda, el graf conceptual 3 defineix que canari és un tipus d'ocell que és groc i canta.

Il·lustració 5. Graf conceptual 1



Il·lustració 6. Graf conceptual 2

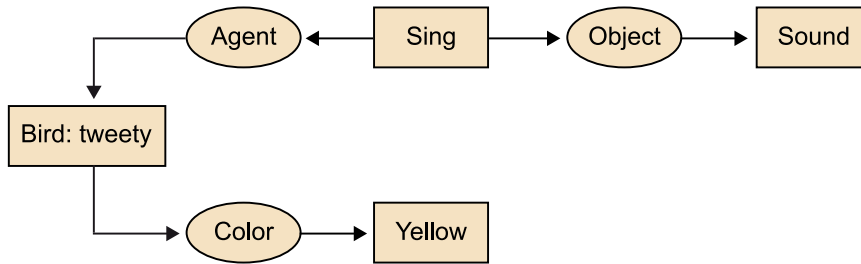


Grafs bipartits i grafs connexos

Recordem que un graf $G = (V, A)$ és **bipartit** si hi ha una partició del conjunt de vèrtexs, és a dir, si es poden separar en dos conjunts disjunts V_1 i V_2 de manera que les arestes només connecten vèrtexs d'un conjunt amb vèrtexs de l'altre.

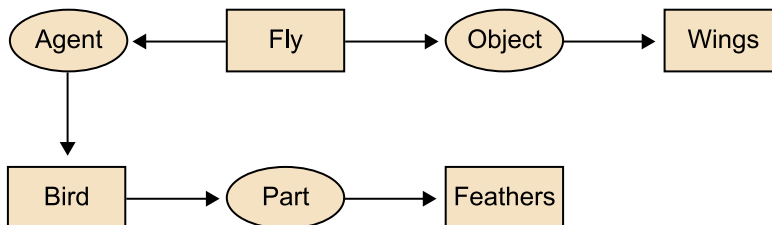
Un graf $G = (V, A)$ és **connex** si per a cada parell de vèrtexs u i v de G , hi ha un camí entre u i v .

Il·lustració 7. Graf conceptual 3



La teoria associada als grafs conceptuals inclou un conjunt d'operacions per a facilitar la creació de nous grafs a partir dels existents. Les operacions són: la còpia, la unió, la restricció i la simplificació. La il·lustració 8 mostra l'operació d'unio entre els grafs conceptuals 1 i 2.

Il·lustració 8. Exemple d'unio dels graf conceptuals 1 i 2

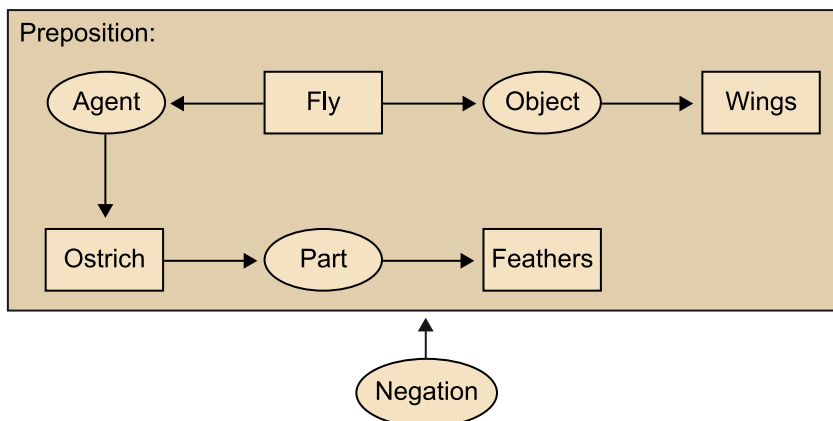


Els exemples vistos de grafs conceptuals demostren com n'és de senzill representar objectes i les seves característiques (en anglès *conjunctive concepts*). En canvi, no hi ha cap mecanisme estàndard establert per a representar la negació o la disjunció. Alguns autors proposen utilitzar la relació unària "negation" associada a una proposició.

Exemple 40. Operació de negació en grafs conceptuals

La il·lustració 9 mostra la proposició "L'estruc no pot volar tot i que té ales".

Il·lustració 9. Operació de negació



2.6.3. La implementació

Els exemples d'esquemes de representació que hem mencionat en aquest subapartat (2.6) són utilitzats de manera molt variada: per a la recollida de coneixement per a la consulta humana, o bé per a la consulta, creació, modificació

o inferència automàtica. Alhora, també, són aplicats en molts dominis com per exemple en l'estudi del llenguatge (bases de coneixement lèxic), la web semàntica o fins i tot la detecció de plagi.

Aquesta varietat d'usos és tan gran que ha provocat la proliferació de molts llenguatges de programació destinats a la implementació d'esquemes de representació. D'aquests llenguatges, cal distingir els que són de **propòsit específic**, com és el cas del llenguatge CycL creat pel projecte Cyc (vegeu el subapartat 1.3.1), dels que són de **propòsit general**, com el Semantic Network Processing System (SNePS) o el Lisp.

Per il·lustrar com s'utilitzen aquests llenguatges, veurem com s'implementa una xarxa semàntica en Lisp. La implementació és senzilla, i a més hi ha diversos estils per a fer-ho.

Exemple 41. Xarxa semàntica implementada en Lisp

El següent codi Lisp és la implementació de l'exemple 37 del subapartat 2.5.1 (xarxa semàntica sobre ocells). Primer de tot, es defineix la funció per a crear relacions d'herència entre conceptes (*entities*) de primer nivell (funció *isa*), i tot seguit per a afegir atributs (funció *attr*):

```
(defun isa (entity1 entity2)
  (setf (get entity1 'isa)
        (cons entity2 (get entity1 'isa))))

(defun attr (entity attribute value)
  (setf (get entity attribute)
        (cons value (get entity attribute))))
```

Tot seguit, codificarem el coneixement propi de l'exemple utilitzant les funcions definides anteriorment:

```
(attr 'animal 'can 'breathe)
(attr 'animal 'can 'move)
(attr 'animal 'has 'skin)
(attr 'bird 'can 'fly)
(attr 'bird 'has 'wings)
(attr 'bird 'has 'feathers)
(isa 'bird 'animal)
(isa 'fish 'animal)
(attr 'canary 'can 'sing)
(attr 'canary 'is 'yellow)
(isa 'canary 'bird)
(attr 'ostrich 'cannot 'fly)
(attr 'ostrich 'is 'tall)
(isa 'ostrich 'bird)
```

Un cop tenim tot el coneixement creat i codificat, dotarem la representació de la funcionalitat per a accedir-hi, on podrem consultar per a cada concepte els seus atributs, fins i tots els que s'hereten.

```
(defun get-attr (entity attr)
  (append (get entity attr)
          (get-from-parents (get entity 'isa) attr)
  )
)

(defun get-from-parents (parents attr)
  (cond ((null parents) nil)
        ((atom parents) (get-attr parents attr))
        (or (get-from-parents (car parents) attr)
            (get-from-parents (cdr parents) attr)
        )
  )
)
```



```

    )
  )
)

(defun entity-attr-value (entity attr value)
  (consp (member value (get-attr entity attr))
  )
)

(defun question-isa (entity parent)
  (consp (member parent (get-attr entity 'isa))
  )
)

```

Ara ja estem en disposició de fer consultes:

- Un canari pot volar?

```
(entity-attr-value 'canary 'can 'fly)
```

- Un ocell té pell?

```
(entity-attr-value 'bird 'has 'skin)
```

- Un canari és un animal?

```
(question-isa 'canary 'animal)
```

També cal destacar la proliferació d'eines específiques per al tractament d'ontologies, fruit de l'augment els últims temps del seu ús, en bona part per l'aplicació a la web semàntica. Veureu més exemples relacionats amb ontologia i web semàntica en el proper mòdul didàctic.

2.7. La representació estructurada

Tot i que la representació estructurada es classifica com un altre tipus d'esquema de representació, no deixa de ser una variant de la representació en xarxa. En concret, la **representació estructurada** és un graf ampliat en què els vèrtexs passen a ser estructures complexes amb una col·lecció de camps anomenats *slots*.

Els *slots* permeten representar valors que poden anar d'un simple número fins a un procediment que realitza una tasca en particular. Els *slots* també poden contenir punters, és a dir, referències a una altra representació estructurada.

Els **marcs** i els *scripts* són exemples d'aquesta representació.

2.7.1. Els marcs

Molts cops el coneixement requereix ser organitzat en unitats més complexes que simples vèrtexs per a que puguin representar situacions o objectes ben coneguts del domini. Aquestes unitats complexes de coneixement s'anomenen *marcs*.

Marc

Minsky (1975) descriu un marc de la manera següent: “Quan algú es troba en una nova situació (o canvia substancialment la seva perspectiva sobre un problema), selecciona des de la seva memòria una estructura anomenada *marc* (*frame*). Un marc és un *framework* guardat en memòria que es pot adaptar per a ajustar-se a la realitat canviant-ne detalls, segons que sigui necessari”.

Segons Minsky els humans treballem amb estructures que recordem de la nostra experiència passada i que ajustem a les noves situacions.

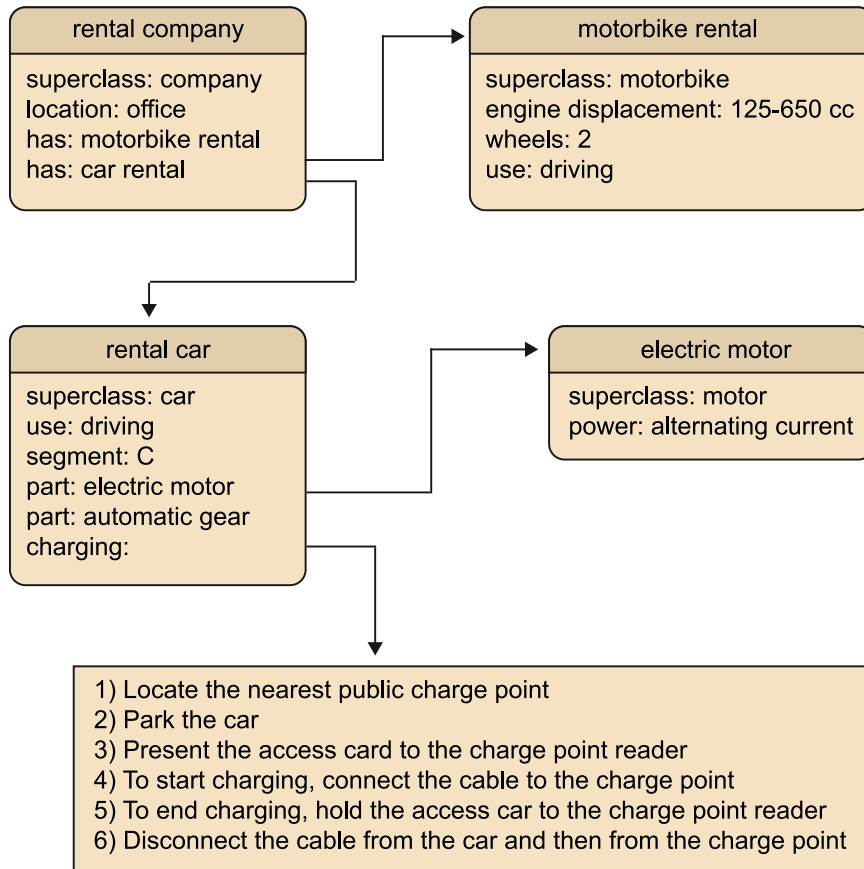
Exemple 42. Definició d'un marc relacionat amb lloguer de cotxes

Qualsevol persona que tingui experiència a conduir un cotxe probablement pot haver conduït més d'un cotxe en la seva vida. Per això si mai n'ha de llogar un, no tindrà cap problema a saber-lo conduir.

Quan l'empresa de lloguer de vehicles fa entrega del cotxe, el llogater espera trobar-hi unes portes d'accés, un seient per al conductor i per a la resta de passatgers, un volant, uns pedals de gas i fre, etc. Però també percep el color del cotxe, la posició del retrovisor i dels comandaments de les llums, etc. A més, recordarà les instruccions de com s'omple el dipòsit de gasolina: accionar la palanca de sota el volant, desenroscar el tap exterior del dipòsit, etc. Tots aquests detalls són organitzats dins d'una estructura conceptual que representa un cotxe genèric.

La figura següent mostra com un cotxe de lloguer pot ser representat per un marc en què es descriuen cadascuna de les característiques. Veiem clar que el cotxe de lloguer és una especialització de la classe *Cotxe*, i inclou *slots* per detallar l'ús, el segment, les parts que el componen i, a més, els passos per a carregar-lo. Recordem que els *slots* poden contenir valors (com ara l'ús, el segment i el canvi), punters a altres marcs (com és el cas del motor elèctric), i també procediments, en aquest cas “com carregar d'electricitat”.

Il·lustració 10. Marc que descriu un cotxe de lloguer



Cada marc individual es pot veure com una estructura de dades que conté la informació rellevant de les seves característiques estereotipades. Els *slots* en un marc contenen informació com la següent:

- Identificador del marc.
- Relacions cap a altres marcs. En l'exemple d'aquest subapartat el cotxe de lloguer és una instància especial de *Cotxe* i l'empresa de lloguer també lloga motos.
- Restriccions d'integritat. Condicions que determinen si un nou objecte encaixa en l'estereotip definit pel marc (principis). Per exemple, la moto de lloguer té una cilindrada entre 125 i 650 centímetres cúbics.
- Informació procedimental per a l'ús de les estructures descrites. És un mecanisme per a incrementar la capacitat expressiva del marc.
- Informació per defecte del marc. El valor d'aquest *slot* es fa servir per defecte quan no s'informa del contrari. Per exemple, la moto té dues rodes.
- Informació d'una instància nova. Seran valors que no estan especificats fins que es doni una instància particular. Per exemple, el *Segment* del cotxe no cal especificar-lo en la definició de *Cotxe*.

Exemple 43. Slot amb informació procedimental

En comptes d'escriure un valor directament en un *slot*, el gràfic següent ens mostra com els *slots àrea* i *perímetre* són procediments que permeten obtenir un valor quan es necessita. Amb aquest mecanisme podem modificar la longitud del rectangle i sempre obtindrem l'àrea associada al nou valor correctament.

Il·lustració 11. Slots amb informació procedimental

rectangle	
superclass:	polygon
length:	8 cm
breadth:	5 cm
area:	$\text{procedure}(x) = \text{length}(x) \cdot \text{breadth}(x)$
perimeter:	$\text{procedure}(x) = 2 \cdot (\text{length}(x) + \text{breadth}(x))$

Per concloure, recordarem que els marcs són útils per a simular el coneixement de sentit comú, coneixement generalment conegut i estereotipat que ens permet detallar totes les característiques típiques del que es modelitza.

La divisió explícita entre relacions i atributs (*slots*) és una propietat avantatjosa que té el marc respecte d'altres esquemes de representació. Cal destacar també la possibilitat de descripció de la seva semàntica a partir de la mateixa estructura.

Però, tot i estar davant d'una representació que proveeix un mètode natural per a representar entitats estereotipades, classes, herència i valors per defecte, G. F. Luger i W. A. Stubblefield avisen dels problemes encara no resolts que presenten els llenguatges de programació per la seva organització. Un d'aquests problemes té relació amb l'herència i es manifesta quan una instància no pot heretar un dels seus *slots* d'una superclasse. En aquest cas l'usuari ha de decidir com s'ha de procedir: ha de triar si cal cancel·lar l'*slot* en concret o només el valor que hauria de contenir.

2.7.2. Els scripts

Scripts

G. F. Luger i W. A. Stubblefield (1998) defineixen l'*script*¹⁴ com: "Una representació estructurada que descriu una seqüència estereotipada d'esdeveniments en un context particular".

Estem davant, doncs, d'una estructura que té com a objectiu la descripció de situacions típiques per mitjà de la representació de seqüències dels esdeveniments que hi succeeixen.

⁽¹⁴⁾En català, *guió*.

Lectura complementària

L'*script* va ser dissenyat originalment per R. Schank i R. Abelson:

R. Schank; R. Abelson (1977). *Scripts, plans, goals and understanding: An inquiry into human knowledge structures*. Erlbaum.

La seva representació és una ampliació dels marcs, on els seus objectes s'estructuren de manera que es pugui donar ordre a les accions que esperem que passin en aquella situació, i els canvis d'estat que s'hi produeixen.

Exemple 44. Ús de *scripts* per a dinar en un restaurant

R. Schank i R. Abelson proposen un exemple molt aclaridor d'ús d'un *script*. Imaginem la seqüència d'escenaris possibles que tenen lloc quan una persona dina en un restaurant:

- Primer la persona entra dins del restaurant.
- Seguidament demana el menjar al cambrer.
- Tot seguit l'hi porten i se'l menja.
- I finalment se'n va del restaurant.

Com hem vist en aquest l'exemple, el conjunt d'escenaris serveix per a ordenar i dividir la seqüència d'esdeveniments que tenen lloc en una situació, però cal, a més, descompondre'ls en més detall, pel que fa a accions i canvis d'estat. Per a fer-ho s'utilitzarà la teoria de **dependències conceptuais**¹⁵, teoria enfocada a representar el coneixement dels esdeveniments que apareixen en les frases en llenguatge natural, i que facilita la inferència de coneixement independentment del llenguatge en què siguin expressades.

⁽¹⁵⁾En anglès, *conceptual dependency*.

La representació de les dependències conceptuais d'una frase es construeix amb el significat que correspon a les paraules que apareixen en la frase. És a dir, es separa el significat de les paraules. Així, tindrem que dues frases amb el mateix significat tindran una única representació. S'utilitzaran les **transicions conceptuais**, formalisme per a modelitzar objectes i accions del món, per a representar de forma única el significat de les frases.

Lectura complementària

Si voleu aprofundir més en la teoria de les dependències conceptuais us recomanem la lectura següent:

R. Schank (1969). "A conceptual dependency parser for natural language". *COLING '69 Proceedings of the 1969 conference on computational linguistics* (pàg. 1-3). Sång-Säby, Suècia.

Exemple 45. *Script* relacionat amb una transició conceptual

Per a descriure l'acció que un client fa en un restaurant per arribar a la taula, hi ha una gran diversitat de frases possibles: "el client caminarà cap a la taula", o també "el client es desplaçarà fins a la taula", etc.

Totes les frases anteriors tenen el mateix significat. Així, per a representar-les, utilitzarem la transició conceptual PTRANS destinada a les accions de moviment:

```
client PTRANS client to table
```

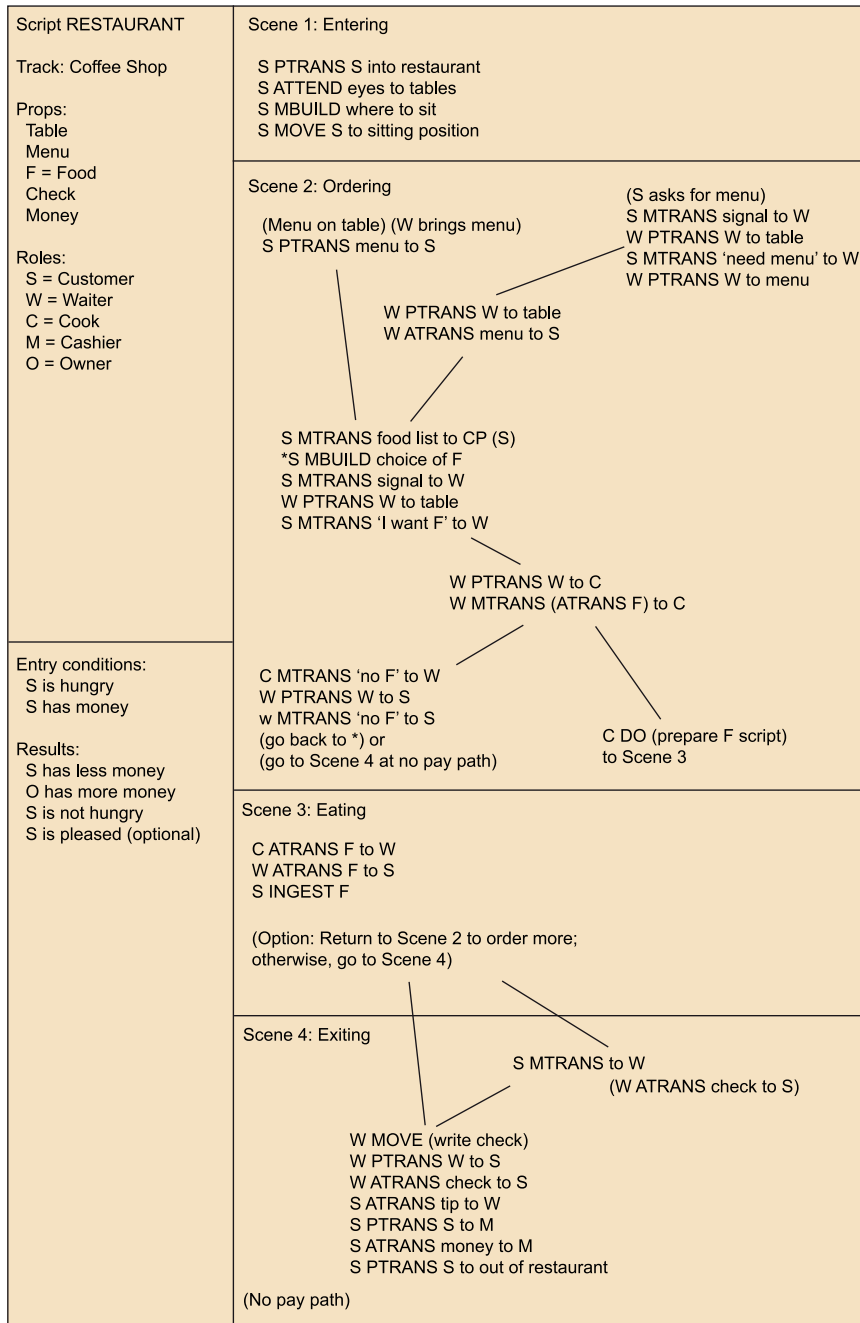
A més de PTRANS, també hi ha MTRANS, utilitzada per a accions de transferència com "donar" o "agafar"; ATRANS, útil per a les accions que representen accions mentals, entre d'altres.

Reflexió

Estem davant d'un tipus de representació canònica del llenguatge natural que pot ser processat pels ordinadors.

Exemple 46. Descripció d'un *script* per a un restaurant

El diagrama següent mostra un *script* que representa un restaurant amb totes les accions que hi passen quan un client hi entra per menjar.



L'*script* es compon dels elements següents:

1) Les **condicions d'entrada**¹⁶, que s'han de satisfer perquè es pugui iniciar l'*script*.

⁽¹⁶⁾En anglès, *entry conditions*.

Exemple 47. Condicions d'entrada en el restaurant

S is hungry
S has money

2) El **resultat**¹⁷; és un conjunt de condicions que són certes quan l'*script* finalitza.

⁽¹⁷⁾En anglès, *results*.

Exemple 48. Resultats en el restaurant

S has less money
 O has more money
 S is not hungry
 S is pleased (optional)

3) Les **propietats**¹⁸; són els objectes involucrats en la situació que es descriu.

⁽¹⁸⁾En anglès abreujades, *props*.

Exemple 49. Propietats en el restaurant

Tables
 Menu
 F = Food
 Check
 Money

4) Els **rols**¹⁹; són les diferents entitats o persones que intervenen en la situació descrita.

⁽¹⁹⁾En anglès, *roles*.

Exemple 50. Rols aplicables al domini del restaurant

S = Customer
 W = Waiter
 C = Cook
 M = Cashier
 O = Owner

5) Els **escenaris**²⁰; són les diferents seqüències d'esdeveniments que tenen lloc en l'*script*.

⁽²⁰⁾En anglès, *scenes*

Exemple 51. Escenaris del restaurant

Scene 1: Entering
 Scene 2: Ordering
 Scene 3: Eating
 Scene 4: Exiting

L'*script* és un mètode de representació a cavall entre la representació estructurada i la procedimental (representació que descrivim en el subapartat següent). Per això avisem que en la bibliografia trobareu alguns autors que el classifiquen d'una manera o d'una altra.

2.8. La representació procedimental

Les arquitectures d'aquest tipus codifiquen el "com fer alguna tasca", i modelitzen així el coneixement procedimental, que ja hem definit. En altres paraules, representa les habilitats que conté el coneixement, com per exemple conduir un cotxe, anar amb bicicleta, etc. En definitiva, la representació procedimental inclou tot el que comporti detallar els passos per a portar a terme una tasca.

Vegeu també

Vegeu la definició de *coneixement procedimental* que proposem en aquesta assignatura en el subapartat 1.3.3 d'aquest mòdul didàctic.

Aquesta és una de les tècniques més utilitzades per a representar coneixement, ja que té l'avantatge que no solament codifica els fets, sinó també la seqüència d'operacions. Però aquest avantatge és també una debilitat, pel fet de tenir intrínscament entrelaçat el coneixement i la seva manipulació.

Podem veure els llenguatges de programació com els mecanismes naturals per a la codificació del coneixement procedimental, a més dels mètodes basats en regles.

Exemple 52. Representació procedimental d'objectes gràfics

La infografia és un dels camps on el disseny gràfic i la informàtica coincideixen per a treballar plegats. La unió de les dues tecnologies ha produït un salt espectacular en el camp de l'animació i el cinema.

En aquest camp, treballar amb gràfics requereix, no solament crear un conjunt estàtic de valors (com ara tots els punts que conformen una imatge en 3D), sinó a més els procediments que en descriuen l'animació.

Aquest tipus de representació permet la generació de complexes formes i moviments, en concret s'utilitzen per a modelitzar fenòmens naturals: fum, plantes, col·lisió o rotació d'objectes, entre d'altres.

Actor/Scriptor Animation System (ASAS) és un llenguatge de programació destinat a la creació d'animacions i gràfics per ordinador. Realment és una ampliació del Lisp, que proveeix una biblioteca de funcionalitats destinades a la manipulació d'objectes animats: vector, polígons, operacions geomètriques, etc.

El codi següent exemplifica com es defineix en ASAS l'animació de la rotació d'un cub:

```
(defop spin-cube-actor
  (param: color)

  (actor (local: (angle 0)
               (d-angle (quo 3 runtime))
               (my-cube (recolor color cube)))
         (see (rotate angle y-axis my-cube))
         (define angle
              (plus angle d-angle))))
```

2.8.1. Les regles de producció

Les regles de producció, altrament anomenades regles *IF-THEN*, conformen una de les tècniques de representació del coneixement més antic. Són un important paradigma de representació aplicat en sistemes experts, planificació automàtica o *l'action selection*. A més, s'ha popularitzat pel seu ús en molts dominis, com ara la medicina, com a mecanisme útil per a diagnosticar patologies mèdiques.

Les regles representen el coneixement utilitzant en el format SI-LLAVORS (*IF-THEN*), format compost per dues parts essencials:

- La part del SI (*IF*), que indica l'antecedent, la premissa, condició o situació.
- La part del LLAVORS (*THEN*), que és el conseqüent, la conclusió, l'acció o la resposta.

Lectura complementària

C. W. Reynolds (1982). "Computer animation with scripts and actors". A: *Computer Graphics* (núm. 16, vol. 3).

Exemple 53. Exemples de regles SI-LLAVORS

SI condueixes un vehicle I s'aproxima una ambulància, LLAVORS has de reduir la velocitat I has de cedir el pas a l'ambulància.

SI la temperatura corporal és superior O igual a 39 C, LLAVORS té febre.

Els avantatges que proporciona aquesta representació són els següents:

- Permet representar coneixement declaratiu i procedimental de manera conjunta.
- És modular, ja que cadascuna de les regles és una porció de coneixement independent.
- És fàcil de manejar: per a modificar el coneixement, només cal afegir o eliminar regles.
- La forma SI-LLAVORS de la regla sovint hi ha una bona correlació amb el llenguatge natural, amb la finalitat de donar explicacions.
- En alguns dominis, els sistemes basats en regles aconsegueixen una semblança del model que les persones utilitzarien per a solucionar elles mateixes el problema.
- Un sistema expert basat en regles pot aconseguir el mateix comportament que un expert humà en alguns dominis.

D'altra banda, també hi ha desavantatges:

- El control de les regles pot esdevenir difús, a causa del fet que les regles poden tenir interaccions entre elles que poden ser difícils de controlar.
- El conjunt de regles no té intrínscament una estructura, la qual cosa fa que sigui difícil la manipulació de grans conjunts de regles.
- No tots els mètodes de resolució de problemes humans són fàcilment representats en aquest formalisme.
- Estem davant d'un mètode que implementat en un sistema expert impossibilita realitzar una prova rigorosa per la validació del seu comportament.

2.8.2. Els programes

Els programes són una representació d'una seqüència d'instruccions, on cadascuna pot ser executada automàticament per a accionar nous esdeveniments o fer canvis d'estat. Com sabeu, l'ordre seqüencial de lectura del codi no serà

el mateix que en l'execució pel fet que hi ha uns mecanismes especials per a indicar el control del flux, control que permet el salt d'un punt a un altre del codi segons l'execució.

2.9. Les ontologies

La definició d'*ontologia* més difosa en l'àmbit de la informàtica és la que va fer Tomas Gruber el 1993. La va definir com una especificació explícita i compartida d'un domini. En altres camps, com per exemple en el de la filosofia, el terme *ontologia* es fa servir des de l'antiguitat. De fet, des que Aristòtil va crear el concepte en l'antiga Grècia. Ara bé, per a entendre bé què és una ontologia, cal analitzar a fons la definició de Gruber. La seva definició parla de "conceptualització explícita, compartida i d'un domini". Estudiem amb calma què volen dir els tres termes d'aquesta expressió:

- "Conceptualització explícita" indica que una ontologia és la representació d'allò que s'ha escrit de manera explícita en algun lloc, és a dir, que no solament és en la ment d'una persona. En aquest sentit, "explícita" també es refereix al fet que ha d'estar codificada en un format que pugui ser compartit i interpretat per sistemes informàtics.
- "Compartida" significa que la representació d'una ontologia no pot ser la visió particular d'algú respecte d'un domini, sinó que ha de representar el coneixement/informació que una comunitat de persones té sobre un domini.
- "D'un domini" indica que està enfocada a representar tan sols una part de la realitat.

Un cop hem clarificat aquests conceptes, podem definir el terme **ontologia** de manera conceptual com la representació de part de la realitat compartida per una comunitat i expressada en un esquema de representació que en permet la interpretació per un sistema informàtic.

Tot i que avui en dia les ontologies acostumen a fer servir esquemes de representació complexes basats en lògica descriptiva o en orientació a objectes, qualsevol representació explícita i consensuada sobre un domini és una ontologia. Així doncs, una llista de conceptes d'un domini en un fitxer de text també podria ser una ontologia, molt simple i amb poques possibilitats d'inferència, però una ontologia de fet.

Segons el seu poder expressiu, les ontologies es poden classificar en **lleugeres** (que fan servir esquemes de representació simples i sense possibilitats d'inferència) o **pesants** (amb esquemes de representació complexos i altes capacitats d'inferència).

La representació ontològica reflecteix la conceptualització del domini entre les persones d'una comunitat, i és necessari un **consens** per a la seva creació. Per aquest motiu una ontologia descriu la conceptualització en termes generals, en lloc de definir situacions específiques.

Tècnicament, els elements que constitueixen una ontologia són els conceptes, les relacions, les instàncies i els atributs:

- Els **conceptes** (vèrtex de la xarxa) representen les categories ontològiques que són rellevants al domini.
- Les **relacions** (arcs de la xarxa) connecten semànticament els conceptes.
- Les **instàncies** (vèrtex fulla de la xarxa) representen els objectes concrets del domini, vèrtex que es classifiquen per mitjà de les relacions amb els conceptes.
- Els **atributs** (combinació de vèrtex i arcs de la xarxa) representen les propietats i els seus valors assignats de conceptes o instàncies. Els valors són alhora altres conceptes o instàncies de la mateixa xarxa.

Molts autors, entre ells T. R. Gruber (1993), comparteixen l'opinió que les ontologies tenen com a propòsit compartir i reutilitzar el coneixement que contenen, a més d'haver de tenir un consens (acord) en el vocabulari utilitzat.

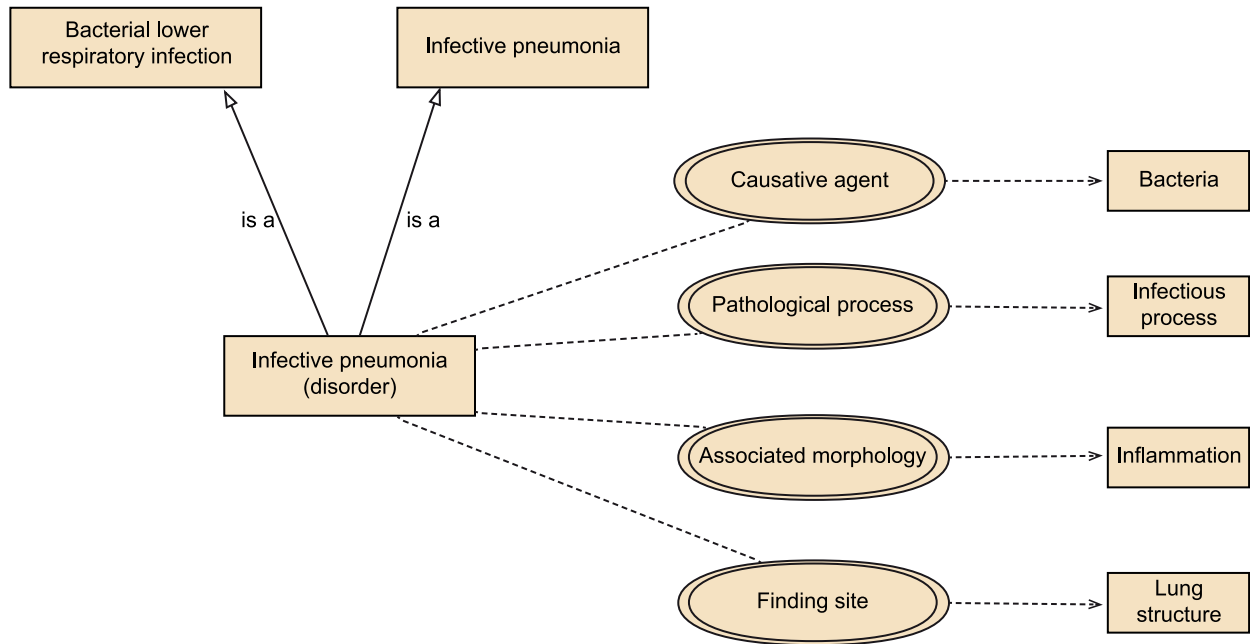
Per aquest fet, són moltes les fonts de coneixement del domini mèdic representades amb ontologies. Un exemple és SNOMED-CT[®], l'ontologia més gran, precisa i multilingüe de terminologia clínica.

Exemple 54. Definició d'un concepte SNOMED-CT[®] i les seves relacions i els seus atributs

En la il·lustració següent mostrem el concepte *infective pneumonia (disorder)* amb totes les seves relacions i els seus atributs.

SNOMED CT

Per conèixer més bé què és SNOMED CT, vegeu el següent recurs d'Internet:
SNOMED Clinical Terms[®]
(SNOMED CT[®]) - US National Library of Medicine

Il·lustració 12. Concepte *Infective pneumonia*

Pel que fa a la visualització, una ontologia s'acostuma a visualitzar com una xarxa semàntica, com hem fet en aquest exemple.

Molt sovint, les ontologies són equiparades amb jerarquies taxonòmiques de classes, però recordem que les taxonomies estan limitades a relacionar de pares a fills o subclasse a superclasse i una ontologia permet modelitzar relacions molt més complexes.

2.10. Qüestions que s'han de tenir en compte

Són moltes les qüestions que sorgeixen quan s'utilitzen alguns dels esquemes anteriorment descrits o tenim la intenció de representar algun tipus de coneixement. Per exemple:

- Quines relacions són les més importants i alhora bàsiques?
- Quin nivell de granularitat hauria de representar?
- Com s'haurien de representar els conjunts dels objectes?
- Donada una gran quantitat de coneixement emmagatzemat, com es pot accedir a les parts rellevants?

Al llarg del mòdul, s'ha donat resposta a algunes d'aquestes qüestions. A continuació, per tal de donar llum a alguna de les respostes, reflexionem sobre certs aspectes estructurals i funcionals.

2.10.1. Les relacions

Hi ha una gran varietat de relacions que conformen el coneixement a partir de relacionar entitats, com ja hem esmentat. Fins ara hem vist relacions d'herència (*és un, tipus de*) i d'instanciació (*instància de*). A part d'aquestes relacions, normalment anomenades *taxonòmiques*, també hi ha altres tipus de relació anomenades *relacions no taxonòmiques*. Aquest segon grup inclou les relacions que no s'utilitzen per a crear la taxonomia d'herència.

Un altre tipus de relació molt comuna és la de meronímia, que inclou les relacions que permeten definir relacions de pertinença. En són exemples les relacions *part of, has, member of, belongs to*, etcètera.

Exemple 55. Exemples de relacions de meronímia

- El Canadà pertany a Amèrica del Nord.
- Un motor és una part d'un cotxe.
- Un arbre té fulles.
- Un dit forma part de la mà.

Altres tipus de relacions utilitzades són les relacions que indiquen propietats semàntiques de la llengua, com per exemple "sinònim" i "antònim".

Exemple 56. Exemples de sinonímia

Alegre és sinònim de *divertit, graciós i rialler*.

Els tipus de relacions que utilitzarem per a representar el coneixement en un problema concret dependran de diferents factors. Les relacions d'herència i pertinença tenen cabuda en gairebé totes les representacions. D'altra banda, les relacions d'instanciació només són necessàries quan la representació ha de contenir instàncies o quan es vol proveir d'exemples. Les relacions que indiquen propietats de la llengua seran molt útils quan es vulgui interpretar o oferir informació en llenguatge natural o quan es vulgui fer tasques d'alineació de conceptes (esbrinar quan dos conceptes són el mateix o no), però altrament poden ser descartables.

Un altre aspecte que convé tenir en compte a l'hora de definir les relacions és l'esquema de representació. Segons l'esquema de representació que s'utilitzi, algunes restriccions d'integritat de les relacions seran molt rellevants o totalment irrelevantes. Per exemple, les restriccions que indiquen que una relació és simètrica, transitiva o funcional prenen molta rellevància si treballem amb esquemes de representació basats en lògica descriptiva, mentre que les relacions de cardinalitat poden no ser tan rellevants. A més no es podran usar

Vegeu també

Recordem que en el subapartat 1.3.6 s'han explicat els tipus de relacions d'herència, objectes i classes, sota l'epígraf "La generalització i especialització del coneixement". A més, en el subapartat 2.6.1 hem entrat a diferenciar què eren els atributs dels objectes i les relacions.

relacions amb una aritat superior a 2. D'altra banda, si utilitzem UML per a representar el coneixement, l'aritat de les relacions podrà ser superior a 2 i les restriccions de cardinalitat esdevenen més rellevants.

2.10.2. Els atributs

Els atributs o propietats són relacions binàries en què un dels participants de la relació és un tipus de dades. Sorprenentment, avui en dia encara hi ha moltes definicions de tipus de dades i no sembla que hi hagi un consens clar sobre la seva semàntica. En aquesta assignatura, per simplificar-ne la identificació, establim els tipus de dades segons aquesta definició: un tipus de dada és una classe els valors de la qual la identifiquen unívocament. Per exemple, el nombre 3 és un tipus de dada perquè dos nombres 3 són el mateix nombre. El mateix passa amb el text "Joan" o amb la data "3 de juliol de 1974". No obstant això, la classe *Llibre* no és un tipus de dades perquè nosaltres podríem tenir dos llibres amb les mateixes dades (ISBN, idioma, títol, autor...), però tot i així ser dos exemplars físics diferents.

Els atributs són molt importants en la representació del coneixement, perquè ens permeten definir les característiques que tenen els elements que modelitzem.

2.10.3. Restriccions d'integritat

Tal com s'ha dit, les restriccions d'integritat són condicions que han de satisfer totes les instàncies de la nostra representació. De restriccions d'integritat n'hi ha moltes, de molts tipus i que afecten diferents elements (classes, objectes, relacions, especialitzacions...).

És important definir les restriccions d'integritat a l'hora de representar el coneixement, perquè aquestes ens permetran definir representacions que tinguin prou informació per a detectar quan un problema donat (una possible instanciació) no pertany al domini representat (no és una instància vàlida de la representació creada). A part d'aquest aspecte pràctic, les restriccions d'integritat ofereixen coneixement intensiu sobre el funcionament i les restriccions dels elements del domini.

Alguns exemples de restriccions d'integritat

Exemples de restriccions d'integritat serien "Tot estudiant de doctorat ha de tenir un títol universitari", "No hi pot haver dos restaurants amb el mateix nom en la mateixa localitat", "L'edat d'una persona ha de ser positiva", o bé "Una persona tan sols pot votar un cop en cada comici electoral".

Les restriccions d'integritat estan suportades de diferent manera en diferents llenguatges de representació de coneixement. Com més complexes siguin les restriccions d'integritat que definim, més complex (i normalment més lent) serà el motor d'inferència necessari per a garantir que se satisfan.

2.10.4. L'elecció de la granularitat

El grau de detall del coneixement representat dependrà del problema i el tipus de coneixement representat, per això s'hauran de tenir en compte les consideracions següents:

- Ser conscients del volum, tant pels fets generals (classes, conceptes) com pels més específics (instàncies, objectes).
- Les entitats més específiques acostumen a tenir un volum més gran, i per tant requereixen més emmagatzemament.
- Els fets més generals acostumen a no ser els adequats per a la inferència de coneixement, ja que tenim menys especificació del domini.

Exemple 57. Elecció de la granularitat

Suposem que estem interessats a representar la informació següent:

“La Mary va tacar en Peter”

Ho podem representar com:

`Tacar (agent (Mary), object (Peter))`

Ara amb aquesta representació és fàcil respondre qüestions com:

“Qui va tacar en Peter?”

Però ara suposem que volem saber:

“La Mary va veure en Peter?”

Per a poder respondre a aquesta pregunta necessitem afegir-hi algun fet més; amb els que tenim, no podem donar-ne una resposta. Escollim generalitzar la implicació que *tacar* també vol dir *veure* el que es taca:

`Tacar (x, y) → Veure (x, y)`

Ara ja estem en disposició d'inferir-ne una resposta i donar-la.

2.10.5. Relacions individuals i col·lectives

Les classes poden tenir relacions que les caracteritzen i les defineixen, però n'hi ha algunes que formen part d'un conjunt més ampli. És a dir, hi ha afirmacions de fets certes a escala de conjunt, que no ho són pas a escala individual.

Exemple 58. Relacions col·lectives

Considerem l'afirmació següent:

“A Austràlia hi ha més ovelles que persones”

El fet que hi hagi més ovelles que no pas persones no és una propietat individual d'una ovel·la, sinó que recau en el conjunt d'elles. Per tant, aquesta propietat s'ha de definir per al conjunt de les ovelles.

Especificar una propietat de manera col·lectiva en una entitat que representa un conjunt d'objectes és molt més eficient que no pas haver-la d'associar a cadascun dels elements del conjunt que representa.

2.10.6. El repte del coneixement incert i heurístic

En l'àmbit de les matemàtiques, és possible enunciar propietats universals i sentències que tenen una validesa absoluta, sense marge de dubte o error. Malauradament, fora d'aquest àmbit el nostre coneixement està subjecte a factors que poden introduir-hi incertesa: errors de percepció, comunicació o interpretació; la subjectivitat dels éssers humans; l'aleatorietat...

Exemple 59. Exemples reals de coneixement incert

En l'àmbit de la física, el resultat de qualsevol experiment porta associat un marge d'error, ja que els aparells de mesura no tenen una precisió infinita.

En una prova mèdica, sempre hi ha el risc d'obtenir un resultat erroni, ja sigui per un error humà, una contaminació de la mostra, la interferència d'una altra malaltia, etc. Per aquest motiu, a vegades es demana repetir una prova, o bé fer-ne d'altres de relacionades per a tenir més fiabilitat en el diagnòstic. Un sistema intel·ligent que treballi en l'àmbit mèdic ha de tenir en compte aquesta possibilitat d'error a l'hora de prendre decisions.

Que un problema tingui associat **coneixement incert** és més normal del que ens pensem, ja que el coneixement és gairebé sempre incomplet o incert. Per aquest motiu hi ha mètodes de modelització especials per a indicar incertesa, que consisteixen normalment a ponderar valors de confiança o pesos: la lògica difusa²¹ n'és un exemple.

⁽²¹⁾En anglès, *fuzzy logic*.

Exemple 60. Plantejament d'un problema amb coneixement difús

Donats els conjunts de ciutats següents:

$X = \{\text{NYC}, \text{Paris}\}$ and $Y = \{\text{Beijing}, \text{NYC}, \text{London}\}$

Com podem representar la idea de “molt lluny” (Klir i Folger, 1988)?

El **coneixement heurístic** és fruit de l'experimentació i és el menys rigorós dels tipus de coneixement. Un exemple de coneixement heurístic seria la regla “una persona no pot fer més de 10 metres”. No hi ha cap fet que limiti que una persona faci més de 10 metres, però, pel que hem observat fins ara, donem aquesta suposició com a certa. Aquest tipus de coneixement conté una subjectivitat que el fa individual.

Alguns autors defineixen el **coneixement heurístic** com “el coneixement que subjau en «l’art d’endevinar bé»”.

Exemple 61. Processament automàtic del llenguatge xinès

En el processament del llenguatge natural, una de les tècniques bàsiques per a extreure informació d’un text és fer una anàlisi de les categories gramaticals de cadascuna de les paraules. És a dir, detectar si la paraula és un nom, verb, adjectiu, article, etc. Després d’aquesta primera anàlisi, posteriorment es poden fer processaments més complexos, com ara una anàlisi sintàctica.

Aquesta anàlisi en un text escrit amb una llengua romànica o germànica és relativament més senzilla que en el cas del xinès. Per posar-ne un exemple, en l’anglès les paraules fan ús de sufixos que n’indiquen la categoria gramatical: *translation* i *translate* utilitzen el sufix *-ion* per al nom i *-e* per al verb. Malauradament, en el xinès les marques flexives que indiquen la categoria gramatical de les paraules són pràcticament inexistent, fet que impossibilita distingir si un mateix ideograma funciona com un nom o com un verb.

Per a resoldre aquest problema, es dissenyen procediments que utilitzen coneixement heurístic basat en regles que compten els patrons amb ponderacions per a la classificació gramatical de cadascun dels ideogrames en funció del context (els altres ideogrames que l’acompanyen en l’oració).

Exercicis d'autoavaluació

1. Considereu el coneixement implicat en els dominis o problemes següents i classifiqueu aquest coneixement com a tàcit, explícit, declaratiu, procedimental, heretable, etc., segons que correspongui. Tingueu en compte que hi pot haver més d'un tipus de coneixement implicat en cada domini de coneixement.

- a) Un arbre genealògic d'una família.
- b) L'existència de Déu.
- c) Dibuixar una casa sobre un paper en blanc.
- d) Saber qui són els hereus d'una persona.
- e) Cuinar una truita de patates.

2. Considereu el context d'un agent de borsa que compra i ven accions. Identifiqueu quins àmbits de coneixement hi estan relacionats, tal com hem fet en l'exemple 1.

3. Considereu l'oració ambigua següent. Utilitzant l'esquema de graf conceptual, representeu-ne els diferents significats possibles.

“En John va veure en David al bosc amb els binocles”

4. Considereu la informació següent i trobeu una xarxa semàntica que la representi:

- a) L'aigua és líquida entre 0 i 100 graus.
- b) L'aigua bull a 100 graus.
- c) L'aigua de l'ampolla d'en Peter està congelada.
- d) Picant és un tipus d'aigua.
- e) Peter té aigua del tipus Picant a la seva ampolla.
- f) Tots els líquids tenen un punt de congelació.
- g) L'alcohol té una densitat de 0,79 grams per centímetre cúbic.
- h) L'aigua té una densitat d'1 gram per centímetre cúbic.

5. Representeu en la lògica de predicats la xarxa semàntica anterior amb el llenguatge Prolog.

Solucionari

1. Descrivim la classificació per a cada domini o problema:

a) És un coneixement que es pot compartir perquè representa fets del món: relació de parentesc entre persones; per tant, el classifiquem com a coneixement explícit i declaratiu. Però, a més, com que aquestes relacions es poden descriure per mitjà de la lògica tradicional, el pare del teu pare és el teu avi, i el pare del pare del teu pare és l'avi del teu pare, també es pot subclassificar com a coneixement inferencial.

b) Ho classificarem com a coneixement tàcit atesa la inexistència de fets en el món real, i la dificultat d'expressar creences per part d'una persona.

c) El coneixement per a dibuixar quelcom és una habilitat apresada de l'experiència, per tant coneixement tàcit. També per a dibuixar sobre un paper cal saber un mètode i seguir uns passos concrets, per tant també ens cal el coneixement procedimental. Però cal, a més, saber les característiques del que es vol dibuixar, en aquest cas una casa, per tant també hi ha coneixement declaratiu.

d) Primer de tot cal saber quines normes regeixen una herència, per tant és un coneixement que es pot comunicar (explícit). Aquestes normes es poden veure com unes regles o fórmules (coneixement procedimental). Finalment, cal saber els fets de parentesc de la persona de la qual es vol conèixer els hereus, coneixement que seria declaratiu.

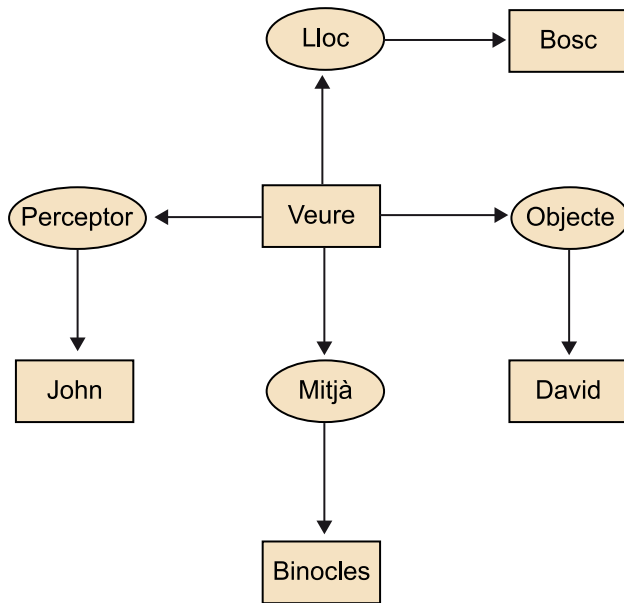
e) Cuinar és un coneixement que s'aprèn amb l'experiència, i per tant és coneixement tàcit. Per a cuinar una truita de patates cal saber-ne els ingredients (coneixement declaratiu) i la recepta, és a dir, la seqüència d'accions que calen per a cuinar el plat (coneixement procedimental).

2. Un agent de borsa necessita el coneixement següent per a portar a terme la tasca de compra i venda de valors:

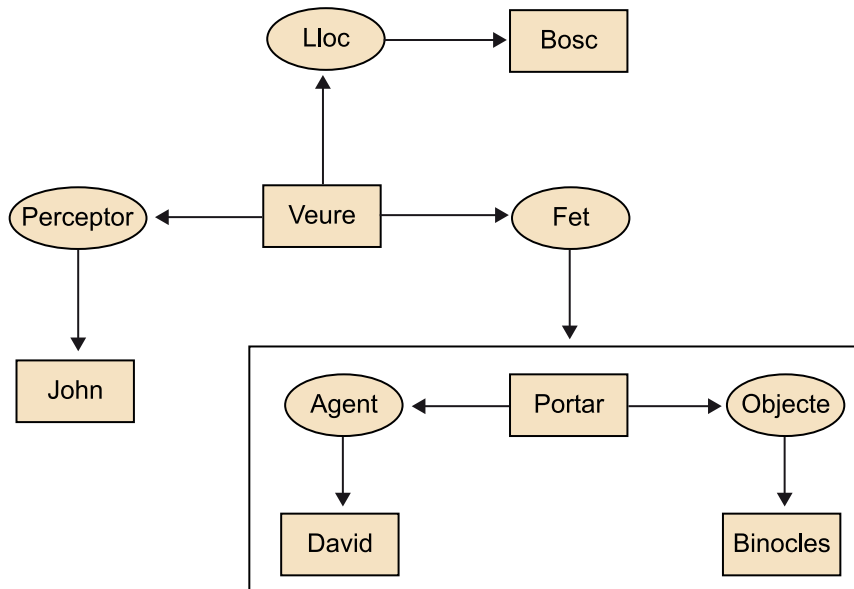
- Coneixement sobre l'economia:
 - Càlcul matemàtic i estadístic per a l'anàlisi de dades.
 - Administració d'empreses, procediments administratius.
 - Estats financers, entendre els diferents tipus de valors i el sistema de comerç.
 - Impostos, lleis reguladores i de protecció, i riscos d'inversions.
- Coneixement sobre sectors empresarials:
 - Sectors emergents, tendències i classificació.
 - Hàbits industrials, farmacèutics i tecnològics.
- Coneixement sobre els mercats financers:
 - Rendes variables nacionals o estrangeres, títols, valors i accions governamentals.
 - Confiança en els mercats de capital.
 - Relació entre la lògica i les emocions.

3. L'oració presenta tres possibles interpretacions que exposem a continuació:

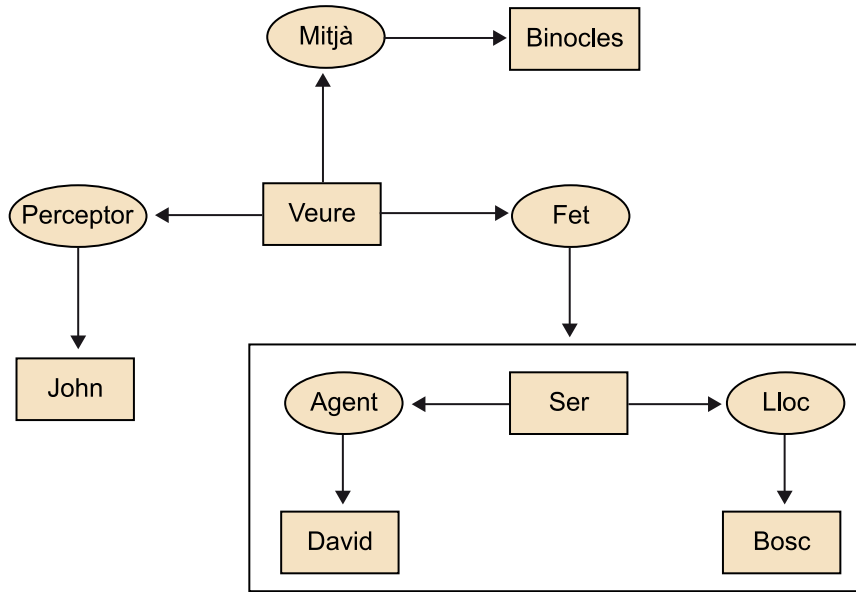
Interpretació 1: en John es trobava al bosc i va veure en David a través d'uns binocles.



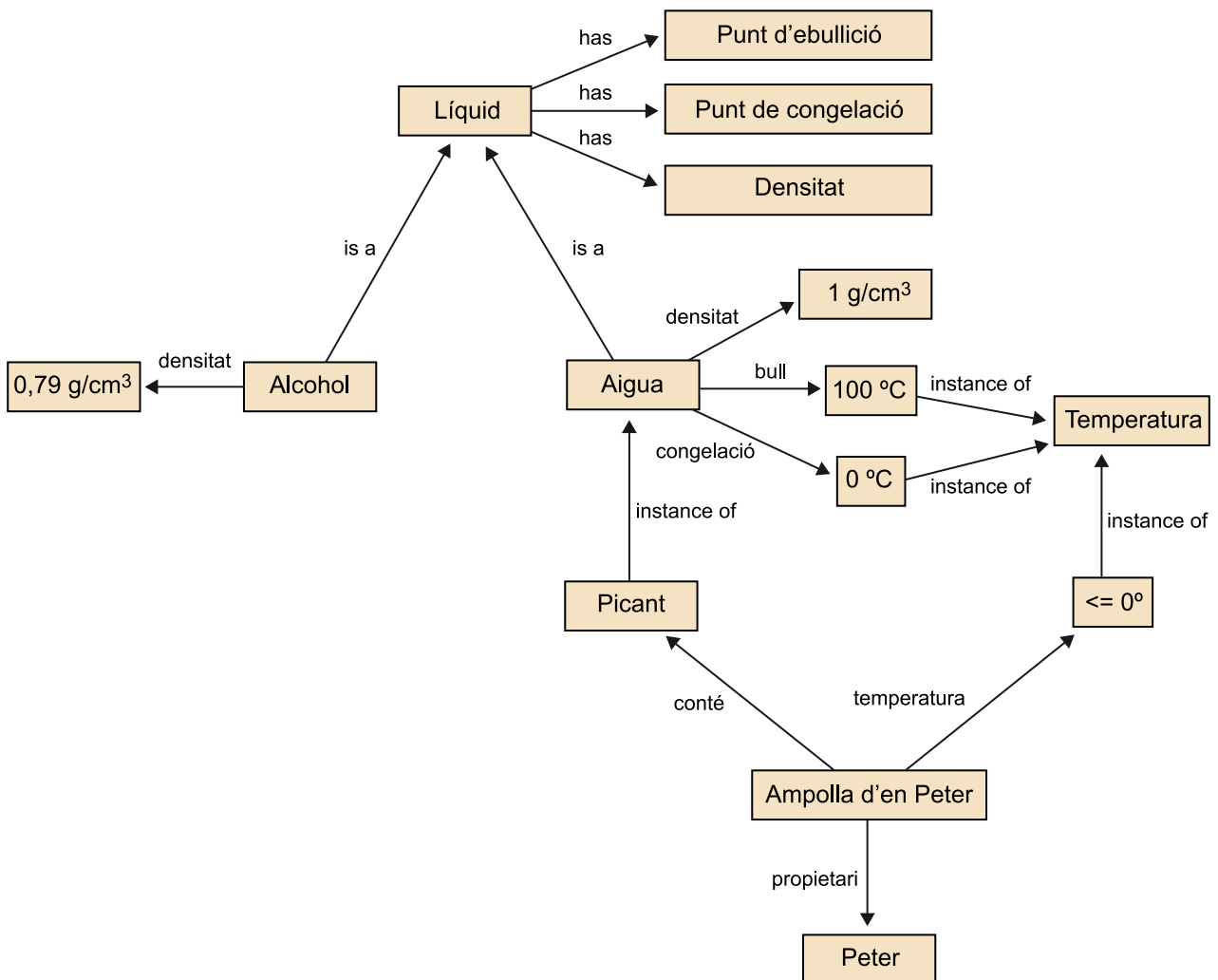
Interpretació 2: en John es trobava al bosc i va veure en David, que portava uns binocles.



Interpretació 3: en John, a través d'uns binocles, va veure en David, que era al bosc.



4. Tot seguit us mostrem una possible xarxa semàntica que representa el coneixement descrit. Heu de tenir present que hi pot haver altres combinacions i variacions de solucions que també són correctes.



5. Us mostrem a continuació una possible representació de la xarxa utilitzant predicats de Prolog. Indiquem la traducció de cada apartat separatament per tal de facilitar-ne la comprensió. Un cop més, tingueu present que hi ha moltes representacions possibles:

a) L'aigua es líquida entre 0 i 100 graus.

```
compound(water).
compound(alcohol).
state(solid).
state(liquid).
state(gas).
freezingPoint(water, 0).
boilingPoint(water, 100).
inState(Object, liquid) :-
    madeFrom(Object, Material),
    temperature(Object, CurrentTemp),
    freezingPoint(Material, FPt),
    boilingPoint(Material, BPt),
    CurrentTemp > FPt,
    CurrentTemp < BPt.
```

Els fets inicials (que no serien necessaris), identifiquen els compostos i els estats de la matèria que es tindran en compte. A continuació definim els punts de congelació i d'ebullició per l'aigua. Finalment, definim la regla que ens marca quan un objecte està en estat líquid: quan la temperatura a la qual es troba l'objecte està entre el punt de congelació i ebullició del material que el compon.

b) L'aigua bull a 100 graus.

```
boilingPoint(water, 100).
inState(Object, gas) :-
    madeFrom(Object, Material),
    temperature(Object, CurrentTemp),
    boilingPoint(Material, BPt),
    CurrentTemp >= BPt.
```

El punt d'ebullició de l'aigua ha estat definit en la sentència anterior (es repeteix només per claredat). Seguidament, definim la regla per tal de concloure que un objecte està en estat gasós, molt similar a la regla per als estats líquids.

c) L'aigua de l'ampolla d'en Peter està congelada.

```
person(peter).
owner(bottle(Content), peter), madeFrom(Content, water), inState(Content, solid).
```

En aquest cas, no donem regles sinó que indiquem uns fets. Utilitzem el predicat *owner* per a descriure el propietari de l'ampolla i *bottle* per a referir-nos a l'ampolla i al contingut. Fixem-nos que tots els fets referits a l'ampolla s'enuncien conjuntament (separats per una coma) per a mantenir el lligam de la variable *Content* entre els tres fets.

d) Picant és un tipus d'aigua.

```
compound(picant).
madeFrom(X, water) :- madeFrom(X, picant).
```

e) Peter té aigua del tipus Picant a la seva ampolla.

```
person(peter).
owner(bottle(X), peter), madeFrom(X, picant).
```

La regla és molt similar a la que hem utilitzat per a la sentència c.

f) Tots els líquids tenen un punt de congelació.

```
freezingPoint(Material, _) :-
    madeFrom(Object, Material),
    inState(Object, liquid).
```

La regla explícita que si un objecte, fet d'un cert material, està en estat líquid, aleshores el material que el compon té punt de congelació (que no indiquem quin és).

g) L'alcohol té una densitat de 0,79 grams per centímetre cúbic.

```
density(alcohol, 0.79).
```

Enunciem aquesta informació com un fet de Prolog.

h) L'aigua té una densitat d'1 gram per centímetre cúbic.

```
density(water, 1).
```

Novament, fem servir un fet per a codificar aquesta informació.

Glossari

aritat *f* En els àmbits de modelització conceptual i de representació del coneixement, nombre de participants en una relació. En el llenguatge matemàtic, l'aritat és el nombre d'arguments o operands d'una funció o operació. Vegeu **relació**.

atribut *m* Característica d'una instància o categoria que pot tenir un determinat valor. Es pot veure també com una relació entre una instància (o categoria) i un tipus de dades.

categoria *f* Abstracció de les característiques que tenen en comú un conjunt d'instàncies.

classe *f* Vegeu **categoria**.

concepte *m* Segons el context, aquest terme es pot utilitzar per a referir-se a una idea abstracta (vegeu **instància**), o bé a un conjunt d'objectes (vegeu **categoria**).

domini *m* Àrea de coneixement que engloba la informació rellevant per a un o més problemes relacionats.

entitat *f* Vegeu **concepte**.

esquema de representació *m* Instrument per a transformar el coneixement d'un domini a un llenguatge simbòlic que pot ser processat computacionalment.

frame *m* Vegeu **marc**.

graf conceptual *m* Esquema conceptual en forma de graf bipartit amb dos tipus de nodes: concepte i relació conceptual. Les arestes indiquen quins conceptes participen en cada relació, permetent representar fàcilment relacions amb múltiples participants. Vegeu **xarxa semàntica**.

granularitat *f* Grau de detall en què es descompon un sistema.

guió *m* Esquema de representació que descriu una seqüència estereotipada d'esdeveniments en un context particular.
en script

individu *m* Vegeu **instància**.

instància *f* Element concret del món real, ja sigui un objecte tangible o una idea abstracta.

instanciació *f* Acció de creació d'una nova instància dins una categoria. Com a part d'aquesta acció, es defineixen uns valors dels seus atributs.

llenguatge formal *m* Llenguatge artificial definit a partir d'un conjunt de convencions i regles que permeten una comunicació precisa. S'utilitza en l'àmbit de les matemàtiques o la lògica.

llenguatge natural *m* Llenguatge utilitzat pels éssers humans per a comunicar-se entre ells les regles del qual deriven de l'ús.

marc *m* Esquema de representació en forma de graf ampliat on cada vèrtex pot tenir associada informació declarativa o procedimental.
en frame

objecte *m* Vegeu **instància**.

ontologia *f* Esquema de representació que especifica de forma explícita i compartida de la informació d'un domini.

relació *f* Vincle existent entre dos o més instàncies o categories. Una relació es caracteritza pel tipus de vinculació, quins són els participants, el paper de cadascun i les restriccions d'integritat (participació obligatòria o opcional, nombre mínim i màxim de participants... Alguns exemples de tipus de vinculació són la generalització/especialització ("is a") o la pertinença (*part of, member of, belongs to, etc.*).

restricció d'integritat *f* Condició que han de satisfer obligatòriament totes les instàncies d'una representació.

script *m* Vegeu **guió**.

sintaxi *f* Conjunt de regles que indiquen com es poden construir i combinar els elements d'un llenguatge per a formar sentències.

semàntica *f* Descripció del significat dels elements d'un llenguatge i la relació entre aquests i el seu referent en un cert domini.

slot *f* Denominació dels atributs en el context dels marcs. Vegeu **atribut**.

xarxa semàntica *f* Esquema de representació en forma de graf etiquetat en què els conceptes es representen com a vèrtexs del graf i les relacions com a arestes en què l'etiqueta indica el tipus de relació. Així doncs, aquest esquema està orientat a representar relacions binàries. Vegeu **graf conceptual**.

Bibliografia

Bibliografia bàsica

Davis, R.; Shrobe, H.; Szolovits, P. (1993). "What is knowledge representation?". *AI magazine* (núm. 14, vol. 1, pàg. 17-33).

Kompridis, N. (2000). "So we need something else for reason to mean". *International journal of philosophical studies* (núm. 8, vol. 3, pàg. 271-295).

Luger, G. L.; Stubblefield, W. (1998). *Artificial intelligence – Structures and strategies for complex problem solving* (3a. ed.). Addison-Wesley.

Nonaka, I.; Takeuchi, H. (1995). *The knowledge creating company: how Japanese companies create the dynamics of innovation*. Nova York: Oxford University Press.

Pole, D.; Mackworth, A. K.; Goebel, R. (1998). "Computational intelligence - A logical approach" (I-XVI, 1-558). Oxford University Press.

Reynolds, C. W. (1982). "Computer animation with scripts and actors". *A: Computer graphics* (núm. 16, vol. 3).

Rich, E.; Knight, K. (2006). *Artificial intelligence*. (105-192). McGraw Hill.

Russell, S.; Norvig, P. (2010). *Artificial intelligence – A modern approach* (3a. ed.). Person Education.

Bibliografia adicional

Rowley, J. (2007, abril). "The wisdom hierarchy: representations of the DIKW hierarchy". *Journal of Information Science* (vol. 33, núm. 2, pàg. 163-180).

Klir, George J.; Folger, Tina A. (1988, gener). *Fuzzy sets, uncertainty and information* (1a. ed.). Prentice Hall.

Sloman, A. (1979). "Epistemology and artificial intelligence". A: D. Michie (ed.). *Expert systems in the microelectronic age*. Edinburgh University Press.

