

UNICAL - A.A. 2006-2007


Gestione della Conoscenza

Prof. Massimo Ruffolo

Ing. Marco Manna

Capitolo 1

- Rappresentazione della Conoscenza
 - Introduzione
 - Linguaggi di Rappresentazione
 - Ragionamento Automatico
 - Sistemi di Rappresentazione
- ...



Rappresentazione della Conoscenza

Introduzione

La Conoscenza

- Un **sistema basato su conoscenza** è un **sistema informativo** in grado di sfruttare l'**informazione** contenuta in una **base di conoscenza** (*knowledge base, KB*) mediante **procedure automatiche di ragionamento**.
- **Conoscenza** = Informazione disponibile per l'Azione (*Dretske, 1981*).
 - Nei computer risiede una grande quantità di informazione ...
 - ... solo una piccola parte di essa può essere sfruttata dagli stessi per agire ...
 - ... poiché la stragrande maggioranza dei file memorizzati nei computer è in linguaggio naturale.

Interrogativo

- È possibile **automatizzare** certe attività che oggi richiedono l'intervento umano?
- È possibile dare una semantica all'informazione in modo tale che i computer possano sfruttarla appieno in applicazioni complesse?
- *La risposta sarà chiara a fine corso!*

Forme di Conoscenza (*Nonaka*)

■ Implicita

- Posseduta dalle persone
- Comunicabile in forma verbale o scritta

■ Tacita

- Presente nelle menti degli individui
- Difficile da comunicare verbalmente (*importante è l'esperienza sensoriale*)

■ Esplicita

- Strutturata (data base, XML+DTD, XML+Shema, ecc.)
- Semi-strutturata (XML, EBCDIC, ecc.)
- Debolmente strutturata (HTML, testi tabulati, ecc.)
- Non strutturata (documenti in linguaggio naturale)

Forme di Conoscenza (*Ryle*)

■ Dichiarativa

- **Conoscere** (*knowing that*) ...
- ... la storia romana
- ... il linguaggio Java

■ Procedurale

- **Saper** (*knowing how*) ...
- ... nuotare (*I'm able to swim*)
- ... programmare in Java (*I can program using Java*)

Conoscenza Dichiarativa

■ Conoscenza **Terminologica**

- **Lessico di una lingua ...**
- ... “madre” **significa** “donna con almeno un figlio”

■ Conoscenza **Nomologica**

- **Regolarità**, leggi generali che regolano il mondo ...
- ... ogni “madre” *sempre* è **più anziana di** “suo figlio”
- ... ogni “madre” *in genere* **ama** “suo figlio”

■ Conoscenza **Fattuale**

- **Fatti particolari ...**
- ... “Marco” è **assistente di** “Massimo”
- ... “Marco” **conosce** “Massimo”

Acquisizione e Conservazione

■ Fonti di Conoscenza

□ Esperienza diretta

- Interazione del soggetto con il suo ambiente

□ Ragionamento

- Deduttivo/inferenza (conclusioni ← premesse)
- Abduittivo (possibili cause ← effetti osservati)
- Induttivo (regole generali ← fatti specifici)

□ Comunicazione

- Uso di sistemi di segni (*in particolare il linguaggio naturale*) per trasferire informazioni da un soggetto a un altro.

■ Funzione della memoria

- **Capacità di Conservare** nel tempo elementi di conoscenza e soprattutto di reperirli con efficienza quando occorre farne uso.

Ruoli della Conoscenza

- **Interpretare *la realtà***
 - Capire che cosa è successo (*ipotizzare le cause di un evento che si è verificato*)
- **Prevedere l'evoluzione *della realtà***
 - Prevedere con approssimazione accettabile quali eventi si verificheranno nel futuro
- **Agire in modo razionale modificando *la realtà***
 - Costruire piani d'azione per raggiungere determinati obiettivi
- **Realizzare Tecnologie informatiche “Evolute”**
 - KDD (Knowledge Discovery in Database)
 - DSS (Decision Support Systems)
 - ES (Expert Systems).

Un po' di Storia

- **Intelligenza Artificiale (1956-1965)**
 - Risoluzione automatica di problemi
- **Rappresentazione della Conoscenza (1965-1975)**
 - Sfruttabile da un programma. *Dichiarativa o Procedurale?*
- **Ingegneria della Conoscenza (1975-1985)**
 - Sistemi esperti. *Dichiarativa + Procedurale.*
- **Crisi e ricerca di strade alternative (1985-1995)**
 - Reti neurali, computazione evolutiva (analisi di segnali, apprendimento automatico, ecc.)
- **Gli agenti e le Ontologie (1995-oggi)**
 - Interoperabilità delle applicazioni e gestione delle risorse in rete.

La centralità dei dati

- Qualsiasi applicazione informatica è costituita da **dati** e **programmi**; le due componenti non hanno però lo stesso peso in applicazioni di tipo differente.
- Vi sono applicazioni centrate maggiormente:
 - sul **processo** (*es. simulazione di sistemi dinamici*)
 - sui **dati** (*es. il web*)
- Le applicazioni centrate sui **dati** sono sempre più diffuse, per cui l'interesse dell'informatica si è gradualmente spostato verso i sistemi centrati sui dati.

L'interoperabilità “interna”

- Per molti anni le aziende hanno sviluppato e utilizzato applicazioni “**chiuse**” (*operano su dati per lo più in formato proprietario*)
 - Difficile interazione tra applicazioni distinte
- Le tecnologie basate su XML offrono un buon livello d'interoperabilità all'interno di una singola azienda
 - Intranet aziendali e integrazione di **dati** e **servizi web** (*web services*)

L'interoperabilità “esterna”

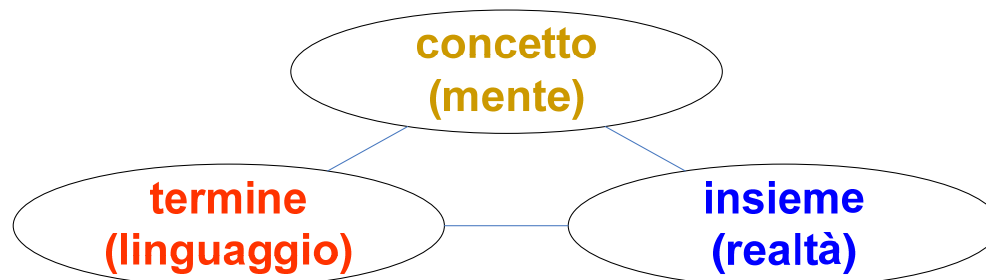
- Le applicazioni “**aperte**” devono essere in grado di interoperare con applicazioni di aziende diverse (ambito extranet)
- Le tecnologie basate su XML sono necessarie ma non sufficienti. XML codifica e standardizza la **sintassi** dei dati ma non la loro **semantica**.
- Due documenti XML possono contenere le due seguenti stringhe "**<spesa>85</spesa>**" "**<costo>85</costo>**".
Un'applicazione informatica non può scoprire che si tratta di due diverse codifiche della stessa informazione, a meno che non sia disponibile la conoscenza del fatto che “prezzo” e “costo” sono **sinonimi** (conoscenza terminologica), almeno nel contesto dell'applicazione in oggetto.

Il triangolo aristotelico (1)

- Assegnamento di un **significato** ai **dati**
- **Significato** → **rappporto** fra *linguaggio* e *realtà*.
 - Il **termine** “*seggiola*” (*sostantivo*) può corrispondere all’*insieme di tutte le seggiole*, oppure a *una seggiola specifica* (un individuo membro dell’insieme delle seggiole).
- La relazione fra *linguaggio* e *realtà* (fra il **termine** “*seggiola*” e l’*insieme delle seggiole concrete*) è detta **relazione semantica**.


Il triangolo aristotelico (2)

- Il rapporto fra **linguaggio** e **realtà** è mediato dai **concetti** (*dalla mente*)
 - La mente è in grado di cogliere gli esatti significati in base al particolare contesto
- Più precisamente, dobbiamo distinguere fra:
 - Il **termine** “**seggiola**” (*“chair” in inglese*)
 - L'**insieme** di tutte le **seggiole** che esistono nella realtà (*l'estensione del termine “seggiola”*)
 - Il **concetto** di **seggiola** così come è rappresentato nella **mente** degli esseri umani (*l'intensione del termine “seggiola”*).



Il triangolo aristotelico (3)

- Il gergo dell'ingegneria della conoscenza spesso confonde i tre vertici del triangolo: così, una definizione di “segiola” sarà indifferentemente denominata “*termine*”, “*concetto*” o “*classe*”.
- Questo **abuso linguistico** è innocuo purché si tenga in mente la distinzione appena introdotta
- A rigore, comunque, nei sistemi informatici sarebbe preferibile parlare sempre di **termini**, dato che un gli oggetti informatici sono comunque **simboli** appartenenti a un **linguaggio formale**.



Rappresentazione della Conoscenza

**Linguaggi di
Rappresentazione**

La Logica Simbolica

■ Problema

- **Rappresentare** la conoscenza in formato **machine-readable** (*un computer può “leggere” tale conoscenza rappresentata e utilizzarla per eseguire compiti d’interesse applicativo*)

■ Soluzione

- **Rappresentazione dichiarativa** tramite **logica simbolica** (*formale*), ed in particolare la **logica dei predicati del primo ordine** (*first order logic, FOL*)

First Order Logic (FOL)

- In **FOL** tutte le rappresentazioni riguardano un **insieme** non vuoto di **individui** detto **universo** (*o dominio*).
- Di questi **individui** possiamo rappresentare **proprietà** oppure **relazioni** che li leghino fra loro. Un **fatto** è dato dal sussistere:
 - di una **proprietà** di un determinato individuo (es. “*Barbara è bionda*”, “*Luigi ha 21 anni*”)
 - oppure di una **relazione** fra più individui (es. “*Alberto è più alto di Barbara*”, “*Alberto ha dato il suo cellulare a Barbara*”, ecc.)

Linguaggi di Rappresentazione

- Un linguaggio per la rappresentazione di conoscenze è un **linguaggio formale**, con sintassi testuale o grafica, le cui espressioni sono utilizzate per rappresentare elementi di conoscenza.
- **Esempio:** rappresentare il significato del termine “madre” come “donna con almeno un figlio”.
 - Linguaggio naturale:
 - $(x \text{ è una madre}) \text{ se e solo se } (x \text{ è una donna ed esiste almeno un } y \text{ tale che } x \text{ è genitore di } y)$
 - First Order Logic (FOL):
 - $\forall x (\text{MADRE}(x) \leftrightarrow \text{DONNA}(x) \wedge \exists y \text{GenDi}(x,y))$
 - Logic Programming (LP):
 - $\text{madre}(\mathbf{X}) \text{ :- donna}(\mathbf{X}), \text{ genDi}(\mathbf{X},\mathbf{Y}).$

Notazione classica per FOL

- Quantificatore Universale
 - $\forall x \dots$ (*per ogni x ...*)
- Quantificatore Esistenziale
 - $\exists y \dots$ (*esiste un y tale che ...*)
- Connettivo Bicondizionale
 - $\dots \leftrightarrow \dots$ (*... se e solo se ...*)
- Connettivo di Congiunzione
 - $\dots \wedge \dots$ (*... e ...*)
- Predicati Monoargomentali
 - **MADRE**(x) (*proprietà su x*)
 - **DONNA**(x) (*proprietà su x*)
- Predicato Biargomentale
 - **GeniDi**(x, y) (*relazione binaria tra x e y*)

Altre notazioni possibili

- Notazione classica per FOL :

- $\forall x (\text{MADRE}(x) \leftrightarrow \text{DONNA}(x) \wedge \exists y \text{GenDi}(x,y))$

- Notazione ASCII per FOL:

- (forall (?x)
 (iff (MADRE ?x)
 (and (DONNA ?x)
 (exists (?y) (GenDi ?x ? y))))))

- Notazione XML per FOL:

- <formula> <forall> <varlist> <var>x</var>
 </varlist> <iff> <atom> <con>MADRE</con>
 <var>x</var> </atom> <and> <atom> <con>DONNA</con>
 <var>x</var> </atom> <exists> <varlist>
 <var>y</var> </varlist> <atom> <con>GenDi</con>
 <var>x</var> <var>y</var> </atom> </exists> </and>
 </iff> </forall> </formula>

```
- <formula>
- <forall>
- <varlist>
  <var>x</var>
</varlist>
- <iff>
- <atom>
  <con>MADRE</con>
  <var>x</var>
</atom>
- <and>
- <atom>
  <con>DONNA</con>
  <var>x</var>
</atom>
- <exists>
- <varlist>
  <var>y</var>
</varlist>
- <atom>
  <con>GenDi</con>
  <var>x</var>
  <var>y</var>
</atom>
</exists>
</and>
</iff>
</forall>
</formula>
```



Rappresentazione della Conoscenza

**Ragionamento
Automatico**

La Deduzione

- Nel contesto in cui ci stiamo muovendo, per “**ragionamento**” s’intende il ragionamento **deduttivo** (*o deduzione*)
- Una **deduzione** è un processo che fa passare da alcune espressioni (dette premesse o ipotesi) a un’espressione (detta conclusione o tesi), in modo tale da conservare l’eventuale verità delle premesse: in altre parole, se le premesse sono vere, lo sarà anche la conclusione.
- Ad esempio, dati come premesse
 1. la **definizione** di “madre”
 2. il fatto che *laura* è una DONNA
 3. il fatto che *laura* è GenitoreDi di *franco*
- si può dedurre come conclusione che
 - *laura* è una MADRE

Deduzione Naturale (calcolo)

■ Premesse

1. $\forall x (\text{MADRE}(x) \leftrightarrow \text{DONNA}(x) \wedge \exists y \text{GenDi}(x, y))$
2. $\text{DONNA}(laura)$
3. $\text{GenDi}(laura, franco)$

■ Deduzione (da 1 per eliminazione di \forall)

4. $\text{MADRE}(laura) \leftrightarrow \text{DONNA}(laura) \wedge \exists y \text{GenDi}(laura, y)$

■ Deduzione (da 3 per introduzione di \exists)

5. $\exists y \text{GenDi}(laura, y)$

■ Deduzione (da 2 e 5 per introduzione di \wedge)

6. $\text{DONNA}(laura) \wedge \exists y \text{GenDi}(laura, y)$

■ Deduzione (da 4 e 6 per eliminazione di \leftrightarrow)

7. $\text{MADRE}(laura)$

Semi-decidibilità di FOL (1)

- Utilizzando FOL si può
 - esprimere conoscenza in modo molto articolate
 - eseguire (*in linea di principio*) in modo automatico ragionamenti complessi.
- C'è però un problema: in FOL la procedura di deduzione non è una procedura di decisione, ma soltanto di **semidecisione**. Ciò significa che:
 - se la conclusione è deducibile delle premesse, la procedura **termina in un numero finito di passi** producendo una prova
 - se la conclusione non è deducibile delle premesse, la procedura **può non terminare**.

Semi-decidibilità di FOL (2)

- La semi-decidibilità di FOL dipende dalla sua notevole espressività:
 - più un linguaggio di rappresentazione è espressivo, più sono problematiche le procedure di ragionamento
- Molte ricerche nel campo dei linguaggi di rappresentazione delle conoscenze hanno l'obiettivo di identificare un sottolinguaggio di FOL tale che:
 - il linguaggio sia comunque abbastanza espressivo per le applicazioni
 - la deduzione si basi su una procedura di **decisione**
 - tale procedura di decisione abbia complessità computazionale accettabile

Le Logiche Descrittive (DL)

- I sistemi di questo tipo hanno preso il nome di **logiche descrittive** (*description logic, DL*). Le DL utilizzano una sintassi semplificata rispetto a FOL.
- Ad esempio, le tre premesse
 1. $\forall x (\text{MADRE}(x) \leftrightarrow \text{DONNA}(x) \wedge \exists y \text{GenDi}(x, y))$
 2. $\text{DONNA}(\textit{laura})$
 3. $\text{GenDi}(\textit{laura}, \textit{franco})$
- in logica descrittiva verrebbero rappresentate come
 1. $\text{MADRE} \equiv \text{DONNA} \sqcap \exists \text{GenDi}$
 2. $\text{DONNA}(\textit{laura})$
 3. $\text{GenDi}(\textit{laura}, \textit{franco})$



Rappresentazione della Conoscenza

**Sistemi di
Rappresentazione**

Le Rappresentazioni (1)

- Esempi di Sistemi di Rappresentazione sono:
 - le basi di dati
 - i diagrammi UML
 - i documenti XML
- Al di là delle ovvie differenze, tutti i sistemi di rappresentazione condividono alcune caratteristiche comuni:
 - **Modello concreto:** conoscenza fattuale su un certo frammento di realtà (es. *“Luigi Rossi ha 10 anni”*)
 - **Modello concettuale:** conoscenza terminologica e nomologica (es. *definire i concetti di persona (avente nome e cognome), di età, ecc.*)
 - **Metamodello:** la specifica degli strumenti formali utilizzabili per definire i primi due (es. *linguaggio FOL, schemi E-R, diagrammi delle classi in UML, ecc*)

Le Rappresentazioni (2)

	<i>basi di dati</i>	<i>specifiche software</i>	<i>logica</i>	<i>RDF</i>	<i>DL</i>
<i>meta-modello</i>	formalismo E-R	diagrammi UML	FOL	Triple RDF	una particolare DL
<i>modello concettuale</i>	un modello concettuale E-R	diagramma delle classi	una teoria del primo ordine	un documento RDFS	una TBox
<i>modello concreto</i>	un'istanza di base di dati	le istanze delle classi	formule atomiche prive di variabili	un documento RDF	un'ABox

Knowledge Representation Systems

- Differenza di fondo tra un **KBS** ed altri sistemi di rappresentazione (*basi dei dati, UML*):
 - nei campi delle **basi di dati** e della **specificazione del software**, i **modelli concettuali** (*E-R, UML, ecc.*) **NON** fanno parte del prodotto finale
 - nei **KBS** i **modelli concettuali** sono parte integrante del prodotto finale in quanto disponibili e utilizzati a runtime dai sistemi software.

L'architettura di un KBS

- **Base di conoscenze (KB):**
 - **Terminological Box (TBox)** per accedere al modello concettuale (*schemi*)
 - **Assertion Box (ABox)**, per accedere al modello concreto (*dati*)
- **Applicazioni Software:**
 - **Servizi di ragionamento** per dedurre conoscenza partendo dalla KB
 - Interfaccia di **editing** per gestire (*lato utente*) i contenuti della KB
 - **Applicazioni software** specifiche
- **Interfaccia di accesso** per la comunicazione tra **KB** e **Applicazioni**

