

Rappresentazione della conoscenza

Logica classica e applicazioni pratiche

- La logica classica si può definire come una lingua franca dell'intelligenza artificiale, ma per le applicazioni pratiche costituisce un linguaggio di riferimento, non uno strumento adatto alla creazione di soluzioni pratiche.
- A partire dagli anni 60 si è sviluppata una branca dell'intelligenza artificiale specificamente orientata alla rappresentazione della conoscenza, che esplora soluzioni alternative alla logica classica, nel tentativo di superarne le limitazioni pratiche.

Limiti della logica classica

- Nonostante la sua precisione con cui permette di esprimere i concetti, la logica classica possiede alcune limitazioni importanti:
 - È caratterizzata da **inadeguatezza espressiva**
 - È **monotona**
 - Presenta svantaggi dal punto di vista **computazionale**

Inadeguatezza espressiva

- Alcuni aspetti dell'inadeguatezza della logica classica possono essere attribuite a differenze con i sistemi cognitivi
- La logica classica è un formalismo **piatto**:
 - tutte le affermazioni si collocano sullo stesso piano
 - esprime conoscenza di carattere generale e immutabile
- Le **procedure di dimostrazione** sono diverse dal ragionamento umano

Svantaggi per la modellazione cognitiva I

- Formalismo **piatto** →
- non incorpora concetti come rilevanza e focalizzazione
 - Importanti per lo svolgimento di compiti in tempo reale
- conoscenza immutabile
- non è adatta a rappresentare informazioni contingenti, soggette a cambiamenti
 - Fatti, non eventi
 - (vedi anche monotonicità)

Svantaggi per la modellazione cognitiva II

- Procedure di **dimostrazione** →
- un sistema logico è consapevole di tutte le conseguenze ultime delle proprie credenze
 - chiusura deduttiva implausibile per sistemi cognitivi (complessità, risorse)
- un sistema cognitivo non contiene contraddizioni
 - consistenza logica è implausibile in un sistema cognitivo

Inadeguatezza espressiva e mondo reale

- Altre limitazioni derivano dalla distanza tra la semantica della logica classica e il mondo reale
- Valori di verità →
- non adatti a rappresentare gli aspetti quantitativi che caratterizzano il mondo reale
- Immutabilità →
- non adatta a rappresentare i cambiamenti che caratterizzano il mondo reale (dinamicità)

Conoscenza del mondo e soggettività

- Il carattere immutabile, certo e privo di gradazioni della rappresentazione del mondo fornita dalla logica classica non si adatta alla relazione tra un sistema cognitivo e il mondo reale, che è mediata dalla percezione
- Non si può assumere certezza nelle conoscenze che derivano dai sistemi percettivi umani o artificiali

Monotonicità

- La logica del primordine è monotona: le conoscenze inserite nel sistema logico non possono essere cancellate.
 - La conoscenza può solo aumentare
- Le nuove conoscenze non possono contraddire quelle già presenti nel sistema:
 - Impossibilità di rappresentare il cambiamento, quindi gli aspetti temporali della conoscenza

Limitazioni computazionali della logica del primordine

- Il calcolo dei predicati è **semi-decidibile**: è possibile dimostrare con certezza cosa discende dalla base di conoscenza
- Non si ha la certezza di poter dimostrare ciò che *non* discende dalla base di conoscenza
- **Inefficienza** nelle procedure di dimostrazione
- Per le sue caratteristiche intrinseche (è un formalismo orientato alla rappresentazione) le procedure di dimostrazione non sono abbastanza efficienti per le applicazioni reali

Logiche non classiche

- Il tentativo di superare i limiti della logica classica ha dato vita a un certo numero di logiche non classiche, che rilasciano alcune caratteristiche della logica classica
- Valori di verità → logiche fuzzy
- Conoscenza certa → ragionamento probabilistico, rappresentazione delle credenze (belief spaces)
- Monotonicità → logiche non monotone (per es. default logics), belief revision, TMS
- Indecidibilità → insiemi decidibili della logica del primordine
- Inefficienza della dimostrazione → uso di euristiche nella dimostrazione automatica

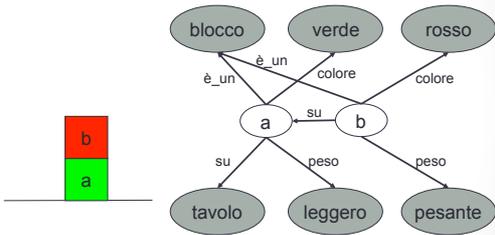
Quadro storico

- Reti semantiche (anni 60-70)
- Altre logiche (anni 70-80)
- Logiche descrittive (anni 80-90)
- Dal 2000: ontologie computazionali

Reti semantiche

- Basate su una struttura reticolare (grafo)
- I nodi del grafo rappresentano concetti
- Gli archi tra i concetti rappresentano le relazioni tra concetti

Esempio



Vantaggi

- Le informazioni relative a un nodo sono immediatamente disponibili
 - Ogni blocco è direttamente collegato alle sue proprietà {peso,colore,su}
- Permette di rappresentare una nozione di rilevanza
 - Dato un focus (blocco a), alcune informazioni si trovano in prossimità

Confronto con rappresentazione logica

blocco (a)
blocco (b)
su (b,a)
su (a,tavolo)
rosso (b)
verde (a)
pesante (b)
leggero (a)

Confronto con rappresentazione logica

blocco (a)
blocco (b)
su (b,a)
su (a,tavolo)
rosso (b)
verde (a)
pesante (b)
leggero (a)

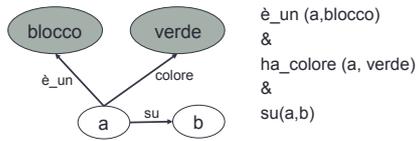
Ragionamento nelle reti semantiche

- Nelle reti semantiche il ragionamento consiste nel seguire un percorso tra nodi
- Su quale blocco si trova il blocco *b*?
- È sufficiente seguire l' arco etichettato "su" nella rete, da *b* verso il nodo a cui punta

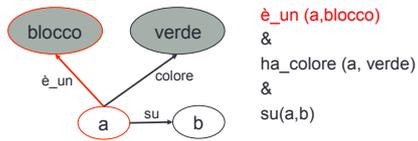
Rapporto con la logica

- La loro espressività corrisponde a un sottoinsieme della logica del primordine
 - Nodi = termini
 - Archi = predicati
- Congiunzione implicita
- Versioni più evolute hanno espressività uguale alla logica del primordine

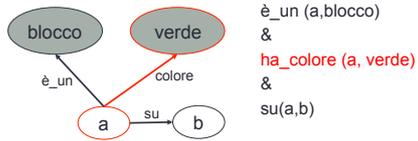
Reti e logica



Reti e logica

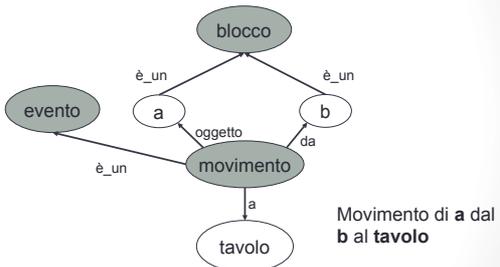


Reti e logica



è_un (a,blocco)
&
ha_colore (a, verde)
&
su(a,b)

Relazioni che coinvolgono più di 2 entità



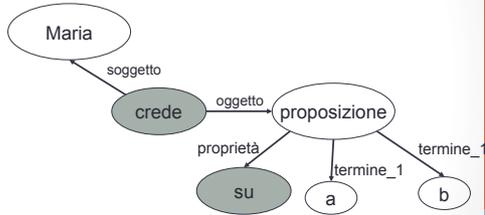
Movimento di **a** dal **b** al **tavolo**

La relazione "movimento" che unisce a, b e tavolo diventa un termine cioè un nodo

Reti semantiche proposizionali

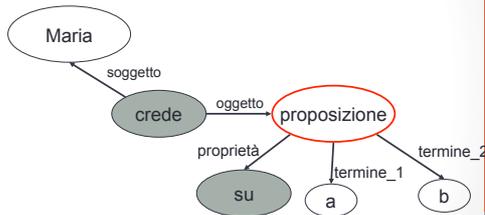
- Le reti semantiche proposizionali includono nodi che rappresentano proposizioni
- Usando nodi per rappresentare proposizioni è possibile introdurre una dimensione epistemica
 - Rappresentare credenze soggettive
 - Lo stesso sistema può rappresentare le credenze di più soggetti senza che insorgano contraddizioni

Rappresentare credenze



“Maria crede che il blocco a sia sul blocco b”
(rappresentazione semplificata)

Rappresentare credenze



“Maria crede **che** il blocco a sia sul blocco b”

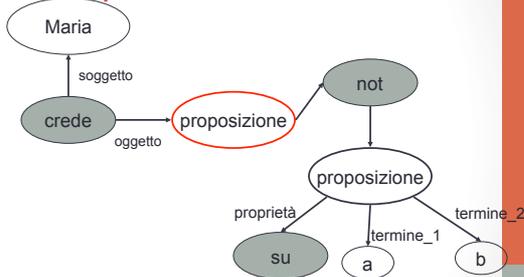
Utilizzi delle reti semantiche

- Utilizzate nell'elaborazione del linguaggio naturale
- La rappresentazione semantica delle frasi viene spesso rappresentata come rete semantica
- Anni '80: utilizzo per dimostrazione di teoremi
- Conoscenza di senso comune: sistema CYC

Rappresentare connettivi

- Tramite le reti proposizionali si possono rappresentare connettivi
 - disgiunzione
 - Implicazione
- Per rappresentare un connettivo si usa un nodo
 - Hendrix (79)

Esempio



“Maria crede che il blocco a non sia sul blocco b”

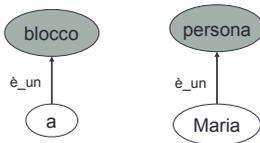
Espressività delle reti semantiche

- Reti semantiche proposizionali possono avere l'espressività della logica del primordine una volta introdotti connettivi, variabili, quantificatori, ecc.
- Anche l'inferenza nelle reti proposizionali ha le stesse caratteristiche che nella logica del primordine
- Soluzione: **limitare l'espressività** privilegiando tipi di ragionamento più comuni e computazionalmente trattabili

Di cosa parla la rete?

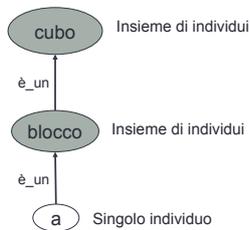
- A livello di rappresentazione, le reti semantiche soffrono di alcune ambiguità
- Eterogeneità:
 - Archi rappresentano relazioni di tipo diverso tra concetti
 - Nodi rappresentano concetti di tipo diverso
- Alcuni tipi di nodi e di archi sono particolarmente importanti

Individui e classi



In bianco, gli individui; in grigio, gli insiemi di individui con caratteristiche uguali, cioè le classi.

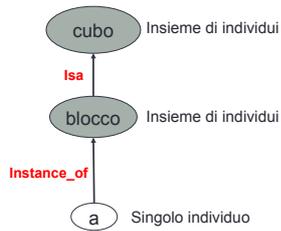
Individui e classi



“What ISA is and isn’ t”

- Gli archi “è un” (IS-A o ISA) hanno un significato diverso se collegano due classi oppure un individuo a una classe
- Brachman (83) propone di distinguere i due tipi di relazioni: “What isa is and isn’ t”.
- Archi **ISA**
 - appartenenza di una classe a una sottoclasse
- Archi **“instance_of”**
 - appartenenza di un individuo a una classe

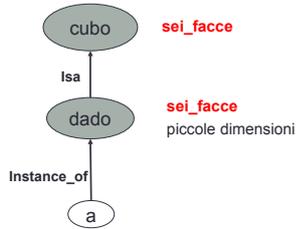
Individui e classi



Tassonomie

- Utilizzando la relazione ISA è possibile esprimere la conoscenza di tipo tassonomico
- Transitività della relazione ISA
- Ragionamento classificatorio
 - Y è una sottoclasse di Z? -> archi ISA
 - x appartiene alla classe Y? -> archi instance_of

Ereditarietà delle proprietà



Perché rappresentare due volte le stesse proprietà?

Vantaggi dell' ereditarietà

- E' possibile avere una rappresentazione meno ridondante facendo una semplice assunzione:
- *Una classe eredita le proprietà delle classi di cui è sottoclasse*
- Se i cubi hanno sei facce e i dadi sono una sottoclasse dei cubi, allora anche i dadi hanno sei facce
- Le sottoclassi hanno proprietà più specifiche

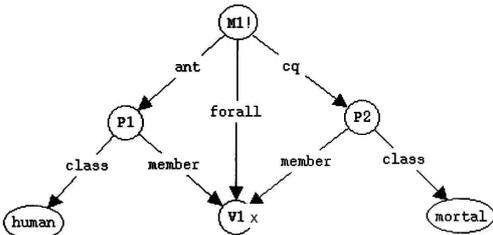
Ragionamento sulle proprietà

- Seguendo il percorso degli archi *isa* e *instance_of* e assumendo l' ereditarietà delle proprietà diventa possibile ragionare sulle proprietà di un individuo / classe
- Quale proprietà ha l' oggetto x?
- L' oggetto x ha la proprietà Y?

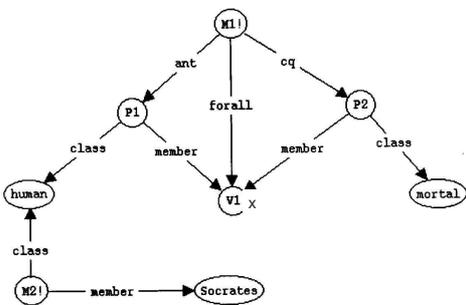
SNePs

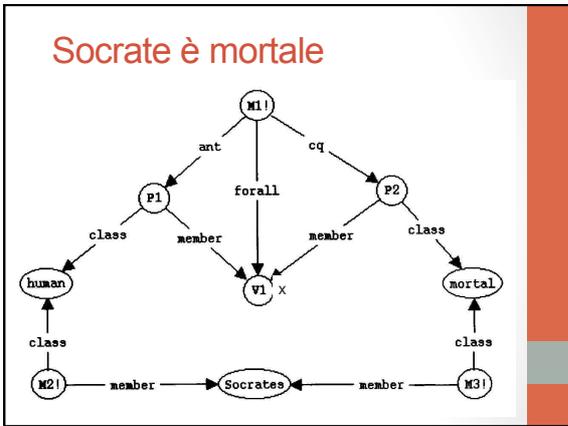
- Rete semantica proposizionale che incorpora alcuni elementi della teoria dei frame (Shapiro 79)
- Rete semantica + motore di ragionamento
- Permette vari tipi di ragionamento:
 - Basato su formule
 - Basato su slot
 - Basato su percorsi

Tutti gli uomini sono mortali

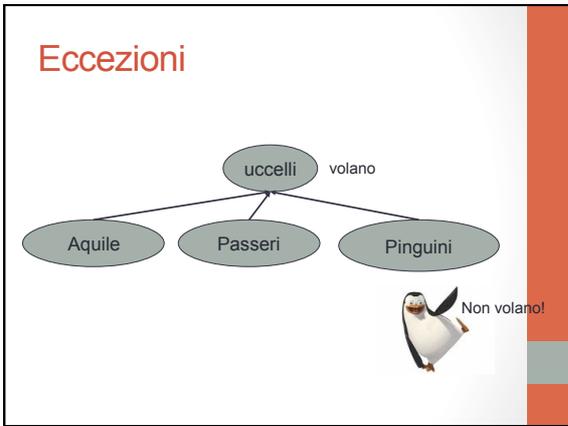


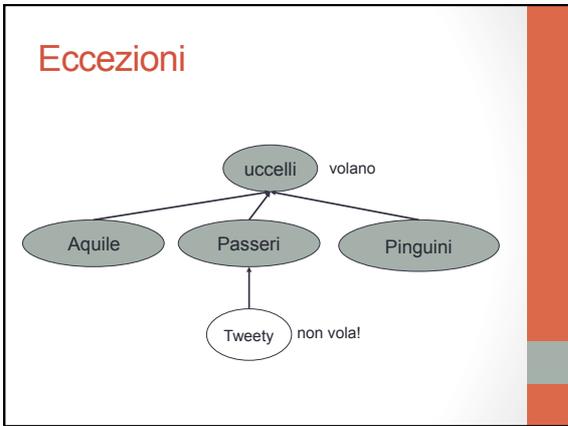
Socrate è un uomo





- ### Bloccare l' ereditarietà
- Le tassonomie esprimono conoscenza generale
 - Nella realtà si incontrano spesso eccezione
 - Per es. gli uccelli volano ma i pinguini *non* volano
 - I canarini volano, ma un certo canarino (Tweety) potrebbe *non* volare





Default rules

- Per gestire le eccezioni è necessario rilasciare la proprietà della monotonicità (conoscenza non diminuisce mai)
- Un esempio di logica non monotona è la **Default Logic**: assunzioni che possono essere cancellate quando sopravviene nuova conoscenza
- Default rules: “gli uccelli *tipicamente* volano”

$D = \{Uccello(X) : Vola(X) \} / \{Vola(X)\}$

se X è un uccello, e si può assumere che voli, possiamo concludere che vola

Ragionamento dato un background

- L'applicazione della default rule dipende dal background
- Il background contiene alcuni fatti:

$W = \{Uccello(Condor), Uccello(Pinguino), non(Vola(Pinguino)), Vola(Aquila)\}$.

- Data la regola precedente, il condor (x = Condor) vola perchè
 - la preconditione Uccello(Condor) è vera
 - la “giustificazione” Vola(Condor) non è logicamente inconsistente con i fatti nel background

Inferenze bloccate dal background

$D = \{Uccello(X) : Vola(X)\} / \{Vola(X)\}$

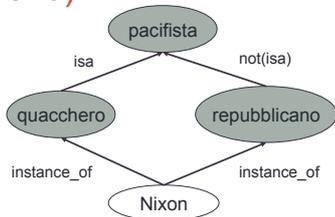
$W = \{Uccello(Condor), Uccello(Pinguino), non(Vola(Pinguino)), Vola(Aquila)\}$.

- Anche se la preconditione $Uccello(Pinguino)$ è vera, la giustificazione $Vola(Pinguino)$ è inconsistente con il background
- Non si può concludere $Vola(Pinguino)$

Trattamento delle eccezioni

- Il trattamento delle eccezioni nelle reti semantiche si basa sul principio che le conoscenze specificate localmente a un certo nodo prevalgono su quelle ereditate
- Un corollario è che le conoscenze che comportano meno passi di inferenza prevalgono su quelle che ne comportano di più
- Questo principio però non permette di scongiurare tutti gli inconvenienti

Ereditarietà multipla (Nixon diamond)



Nixon è pacifista o guerrafondaio? In quanto quacchero, lo è; in quanto repubblicano, non lo è.

Nemmeno la logica dei default risolve il problema, ciascun default blocca l'altro

Frame

- Ispirazione: articolo-manifesto di Marvin Minsky (1975)
- Evoluzione delle reti semantiche finalizzata a rappresentare la conoscenza di tipo *stereotipato*
- Conoscenza di sfondo utile per alcune applicazioni come la visione artificiale o l'elaborazione del linguaggio

Definizione di frame

- A frame is a data structure for representing **stereotypical situations**, like being in a certain kind of living room or going to a child's birthday party. (...) The top levels of the frame are **fixed**, and represent things that are always true about the supposed situations. The lower levels have many terminals - **slots** that must be filled by specific instances or data

Slots

- Each terminal [slot] can specify **conditions** its assignments must meet. (The assignments themselves are usually smaller subframes).
- Conditions may require [...] a terminal assignment to be a person, an object [...] or a **pointer to a subframe** of a certain type
- A frame's terminals are usually filled by **default** assignments
- Collections of related frames are linked together in a **frame system**

Frame: struttura

- Identificativo
- Slot: frame più generali / più specifici
 - Questi slot permettono di creare una tassonomia di frame
- Slot
 - I valori degli slot sono vincolati a un certo tipo
 - Età, anno, nomi
 - Il contenuto di un frame può puntare a un altro frame
- Procedure per il calcolo automatico dei valori
- Valori predefiniti

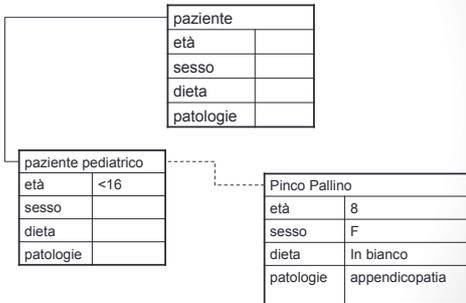
Esempio: il frame del ristorante

- Sottotipo di: frame attività commerciale tassonomia
- Tipi specifici: tassonomia
 - Self-service, trattoria, caffetteria
 - Default: trattoria
- Indirizzo: slots
 - Range: indirizzi
 - If-needed: guarda sul menu procedure di calcolo valori
- Nome:
 - Range: nomi
 - If-needed: guarda sul menu
- Action:
 - script del ristorante (entra, siediti, ordina, mangia, ecc.)
- Tipi di cibo
 - Range: etnico, tradizionale, pesce, fast food
 - Default: tradizionale

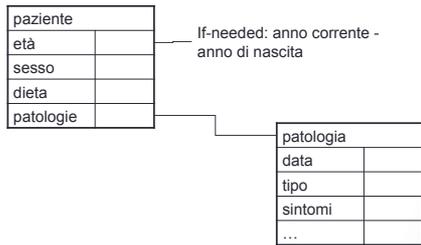
Sistemi basati su frame

- I sistemi di rappresentazione della conoscenza basati su frame sfruttano il sistema di ereditarietà e *default* presenti nei frame.
- L' *attaccamento procedurale* consiste nell' associare agli slot procedure per calcolarne il valore (*if-needed, if-added*)

Ereditarietà: esempio



Collegamenti tra frame: esempio



Regole di produzione

- Conoscenza di tipo condizione-azione
- Regole IF - THEN (“produzioni”)
 - Parte sinistra: condizionj per l’ applicazione delle regola
 - Parte destra: azionj che sono effettuate se la condizione è soddisfatta

Applicazioni

- Conoscenza dichiarativa
 - IF allatta(animale,prole) THEN isa(animale,mammifero) [classificazione]
 - IF febbre>38,5 and tosse THEN malattia_da_raffreddamento [diagnosi]
- Conoscenza procedurale
 - IF intensità(sibilo)>t THEN chiudi(valvola,5mm)

Sistemi a regole

- Base delle regole
- Memoria di lavoro
- Interprete delle regole
- Interprete delle regole esegue ciclo:
 1. Confronto (match) tra fatti (nella memoria di lavoro) e regole
 2. Risoluzione di conflitti (più regole applicabili)
 3. Esecuzione (e aggiunta di nuovi fatti alla base)

Conflitti: criteri

- Refrattarietà: la stessa regola non viene applicata più volte
- Recenza: vengono privilegiate le regole che si applicano ai fatti più recenti
- Specificità: vengono applicate le regole più specifiche
- Pesatura: assegnamento di importanza (*peso*) a priori alle regole o ai fatti

Diagnosi

- Ragionamento abduittivo
 - malattia da raffreddamento -> febbre e dolori
 - IF febbre e dolori THEN malattia da raffreddamento
- Ragionamento non monotono
 - IF febbre e dolori e esantema -> morbillo
- Conoscenza incerta
 - Probabilità associate ai fatti: Esantema 75%
 - Probabilità associate alle regole

Ragionamento in presenza di incertezza

- Sistema esperto Mycin sviluppato a Stanford nei primi anni 70 (Shortliffe, 75)
- Fatti associati a probabilità:
(IDENTITY ORGANISM-1 PSEUDOMONAS 0.8)
(IDENTITY ORGANISM-1 E. COLI 0.15)
(SITE CULTURE-2 THROAT 1.0)

Mycin: esempio di regola e esempio di spiegazione

PREMISE: (AND
(SAME CNTXT GRAM GRAMNEG)
(SAME CNTXT MORPH ROD)
(SAME CNTXT AIR ANAEROBIC))
ACTION: (CONCLUDE CNTXT IDENTITY BACTEROIDES
TALLY .6)

- IF: 1) The site of the culture is blood, and 2) The gram stain of the organism is gramneg, and 3) The morphology of the organism is rod, and 4) The portal of entry of the organism is udne, and 5) The patient has not had a genito-urinary manipulative procedure, and 6) Cystitis is not a problem for which the patient has been treated THEN: There is *suggestive evidence* (.6) that the identity of the organism is e.coli

Aspetti cognitivi

- Base delle regole = memoria a lungo termine
- Memoria di lavoro = memoria a breve termine
- Simulazione cognitiva:
 - Restrizioni su memoria di lavoro (7+/-2 Miller 56)
 - Memoria di lavoro temporanea (schema FIFO)
 - Restrizioni su match (completo e sequenziale)
 - Dimensione della base
 - Memoria di lavoro
 - Complessità degli antecedenti nella base
- Esperimenti su memoria a breve termine (Anderson 76)

Dichiarativo vs procedurale



Produzioni oggi: Jess Rule Engine

- Jess is a rule engine and scripting environment written entirely in Java language
- Jess is small, light, and one of the fastest rule engines available
- Jess uses an enhanced version of the Rete algorithm to process rules.
 - Rete is a very efficient mechanism for solving the difficult many-to-many matching problem
