

MARKUP PROCEDURALE e MARKUP DICHIARATIVO

La distinzione tra **procedurale** e **dichiarativo** è una distinzione introdotta a proposito (a) dei sistemi di *rappresentazione della conoscenza* e (b) dei *linguaggi di programmazione*.

(a) Secondo lo psicologo cognitivo John Robert Anderson (*Language, memory, and thought*, Hillsdale, N.J., Erlbaum Associates, 1976) – che si richiama all'impostazione simbolica proposta da Allen Newell (*Human problem solving*, con Herbert A. Simon, Englewood Cliffs, N.J., Prentice-Hall, 1972), uno dei padri riconosciuti dell'intelligenza artificiale – possiamo ripartire le conoscenze umane in due tipi di rappresentazioni tra loro irriducibili: le rappresentazioni *dichiarative* e le rappresentazioni *procedurali*. Le conoscenze dichiarative possono essere espresse con asserzioni vere o false, mentre le conoscenze procedurali sono quelle su cui si basa l'esecuzione di un certo compito: una conoscenza *dichiarativa* è conoscere 'che,' mentre una conoscenza *procedurale* è conoscere 'come.' I modelli di rappresentazione dichiarativi fanno riferimento a rappresentazioni di oggetti ed eventi e alle relazioni tra tali oggetti ed eventi ed altri oggetti ed eventi; essi si concentrano sul 'perché' piuttosto che sul 'come.' I modelli di rappresentazione procedurali, invece, si concentrano sui compiti che debbono essere eseguiti per raggiungere un particolare obiettivo; essi considerano il 'come' piuttosto che il 'perché.'

(b) La programmazione *procedurale* è spesso assimilata alla programmazione *imperativa* che, a sua volta, si distingue nettamente dalla programmazione *dichiarativa*:

i programmi *imperativi* [o *procedurali*] forniscono una specificazione esplicita dell'algoritmo che porta ad un certo risultato, mentre i programmi *dichiarativi* presentano una specificazione esplicita del risultato, affidando l'implementazione dell'algoritmo al software d'appoggio [imperative programs explicitly specify an algorithm to achieve a goal, while declarative programs explicitly specify the goal and leave the implementation of the algorithm to the support software], *Wikipedia* (English), *sub voce* Declarative programming, http://en.wikipedia.org/wiki/Declarative_programming.

Si parla quindi, rispettivamente, di linguaggi di programmazione *dichiarativi* e di linguaggi di programmazione imperativi o *procedurali*: un linguaggio dichiarativo specifica 'che cosa' si vuole ottenere, un linguaggio procedurale specifica 'come' lo si può ottenere; un linguaggio dichiarativo formula asserzioni vere o false, un linguaggio procedurale formula istruzioni o comandi.

Conformemente a quest'uso, a proposito dei *linguaggi di markup*, nell'opuscolo *Codifica elettronica dei testi letterari*, curato dall'Archivio Digitale della Letteratura Siciliana DigiSic

http://ww2.unime.it/digisic/documentazione/codifica_baltico/Digisic.pdf

correttamente si afferma:

I linguaggi procedurali, conosciuti anche come *specific markup language*, sono costituiti da istruzioni operative che, analogamente ai programmi, eseguono delle procedure computazionali sulle porzioni di testo cui fanno riferimento. I linguaggi dichiarativi, detti anche *generic markup language*, invece, non attivano con i loro tag una procedura informatica, ma piuttosto specificano, dichiarano, la funzione astratta assolta dal blocco di testo cui si riferiscono. (p. 17)

Siccome la loro caratteristica essenziale è quella di fornire istruzioni,

i linguaggi procedurali vengono abitualmente usati per codifiche di tipo presentazionale, finalizzate alla definizione precisa della resa grafica del documento. (p. 17)

Quando però si afferma che

mentre i linguaggi di codifica procedurale dicono al computer come fare qualcosa, ossia come rendere graficamente un testo, i linguaggi di *markup* generico dicono al computer cosa è qualcosa, identificano la funzione strutturale dei blocchi testuali, (p. 19)

ne può nascere una certa ambiguità. Infatti, poiché si dice che i linguaggi di *markup* procedurali hanno a che fare con l'aspetto grafico, mentre i linguaggi di *markup* dichiarativi hanno a che fare con la struttura logica, talvolta si giunge ad affermare che i primi si applicano al *documento*, mentre i secondi si applicano al *testo*. Ma il passaggio è indebito, perché se si considera bene la questione, è facile rendersi conto che entrambi i tipi di linguaggio si riferiscono a stringhe di caratteri codificati, ossia a documenti e non al testo di cui costituiscono un'espressione, o un' 'immagine': «il testo non ha una natura materiale» e del testo abbiamo «sempre» e «soltanto un'immagine» (C. Segre, *Avviamento all'analisi del testo letterario*, Torino, Einaudi, 1985, p. 378).

La struttura che un linguaggio di *markup* assegna alla stringa di caratteri codificati è dunque e sempre la struttura logica del documento e non la struttura logica del testo.

D. Buzzetti, 18 ottobre 2007.