Verso una nuova era: i linguaggi di codifica

L'orientamento prevalentemente matematico dei primi *software* e applicazioni dell'Informatica Umanistica hanno favorito la diffusione dell'idea che i calcolatori siano esclusivamente delle macchine capaci, nel contesto della ricerca storica, di eseguire complessi calcoli statistici su imponenti moli di dati. E sebbene, accantonate le proprie pretese di esclusività scientifica, le metodologie di quantificazione supportate dall'utilizzo di *database* siano comunque entrate a pieno titolo nella cassetta degli attrezzi dello storico, a lungo è rimasto aperto il problema – ben più rilevante – di «come rispettare le caratteristiche della fonte nella sua integrità nel momento in cui si doveva costringerla nella camicia di Nesso di un programma rigidamente organizzato, e come stabilire collegamenti tra fonti diverse inerenti alla stessa ricerca»[1]. Tra i rappresentanti più autorevoli della diffusa diffidenza nei confronti dell'uso del computer in analisi storiche Alessandro Pratesi, nel corso dell'ormai storica *Table Ronde CNRS* organizzata dall'École Française di Roma e dall'Istituto di Storia Medievale di Pisa, aveva infatti denunciato nel 1975 che risposte soddisfacenti da un trattamento improntato all'informatica delle fonti documentarie medievali, si sarebbero potute conseguire soltanto con una memorizzazione dei documenti *in extenso*; sulla stessa scia Ermanno Califano aveva sottolineato come l'alternativa tra *full-text* e immissione di dati significativi in un *database* rappresentasse una scelta fondamentale tra informazione globale e diretta ed informazione preselezionata da altri, senza possibilità di un raffronto immediato con il documento originario[2]. Il concetto è stato ribadito, negli anni Novanta, da Joacquim Carvalho:

> I migliori metodi per l'*input* dei dati forniti da fonti storiche sono quelli che preservano la struttura originaria dell'informazione; un'unica fonte dovrebbe essere registrata come un unico *file*; la successione dei diversi elementi di informazione nel *file* dovrebbe seguire fedelmente la successione con cui sono riportati nella fonte originaria[3].

[1] S. SOLDANI, L. TOMASSINI, *Lo storico e il computer,* in *Storia & Computer. Alla ricerca del passato con l'informatica* cit., pp. 1-28:10.

[2] Cfr. A. PRATESI, *Limiti e difficoltà dell'uso dell'informatica per lo studio della forma diplomatica e giuridica dei documenti medievali,* in *Informatique et Histoire Médiévale.* Communications et débats de la Table Ronde CNRS, organisée par l'Ecole française de Rome et l'Institut d'Histoire Médiévale de l'Université de Pise (Rome, 20-22 mai 1975), presentés par L. FOSSIER, A. VAUCHEZ, C. VIOLANTE, Roma, Ecole française de Rome 1977 (Collection de l'Ecole française de Rome, 31), pp. 187-190; E. CALIFANO, *Registrazione diretta e integrale dei documenti. Utilizzazione di regesti,* in *Informatique et Histoire Médiévale* cit., pp. 253-256:254.

[3] J. CARVALHO, *Soluzioni informatiche per microstorici,* in *Quaderni Storici,* ns. 78 (1991), *Informatica e fonti storiche,* pp. 761-791:777.

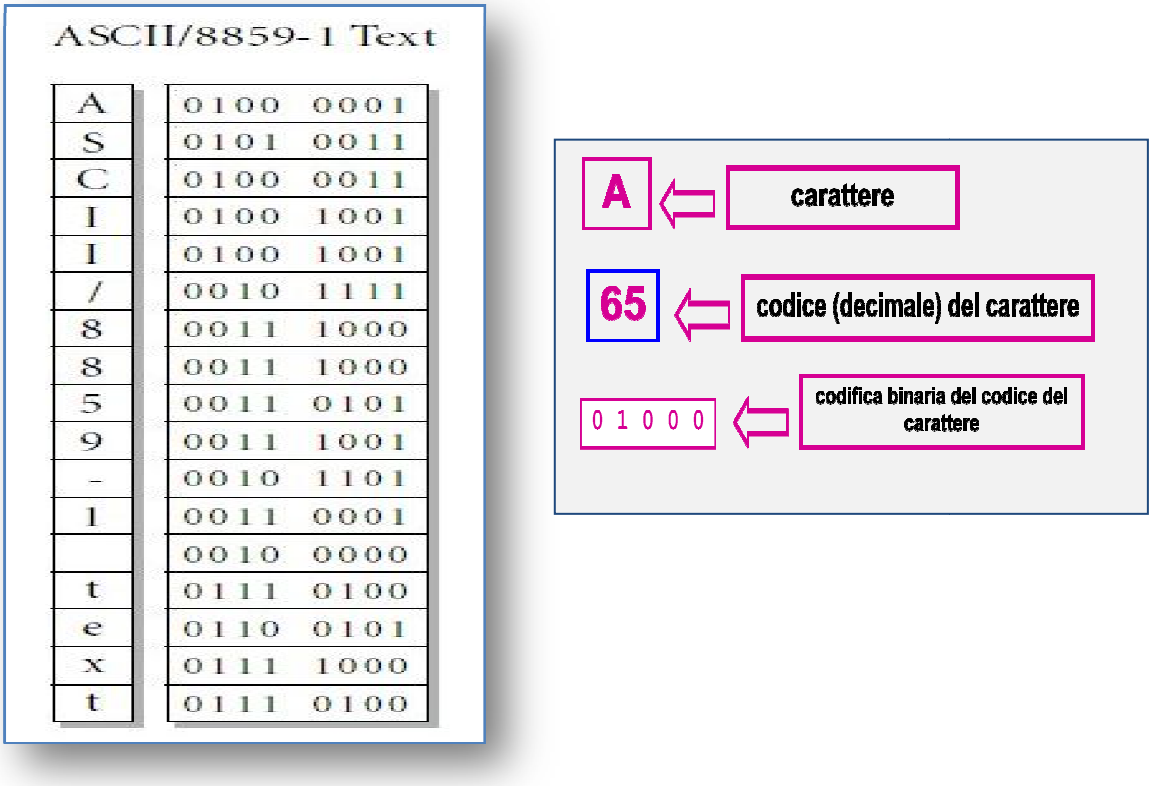
Con queste affermazioni i tre studiosi davano voce all'avvertita esigenza metodologica di rispettare la natura eminentemente contestuale dell'informazione contenuta nelle fonti storiche e, parallelamente, essere in grado di valutare il contesto di produzione del documento, senza tralasciare nessun elemento, nessun dato utile a confermare le ipotesi di ricerca proposte. Queste velleità – lo ha sottolineato Robert Rowland – trovavano giustificazione nell'impossibilità, propria dello storico, di formulare ipotesi di lavoro se non dopo un'analisi preliminare dei dati[4]. La tavola rotonda romana evidenziava dunque all'ordine del giorno la consapevolezza di come nell'approccio alle fonti storiche, sensibili al contesto e collegate le une alle altre, ogni informazione estratta e normalizzata in un *database* rischiasse di perdere elementi utili a renderla pienamente intellegibile e correttamente interpretabile e che un vero *plus*-valore dato dal trattamento informatico potesse essere raggiunto solo elaborando un modello di rappresentazione della fonte che consentisse l'utilizzo dei dati senza impoverirne o alterarne i molteplici significati, conservandone sfumature e ambivalenze. L'approccio proposto dallo stesso Pratesi non chiedeva più al computer di reperire nei testi dati quantitativi da sottoporre ad elaborazione, ma piuttosto di esplorare le strutture informative in essi presenti, recuperarle, riorganizzarle e aggregarle secondo i punti di vista suggeriti dalle ipotesi di ricerca, attivando o evidenziando connessioni prima sconosciute o scarsamente evidenti ma – al contempo – mantenendone l'integrità. Nell'utilizzo dei *database* il computer mostrava i quei limiti che le analisi storiche – finalizzate «all'interpretazione dell'insieme dell'insieme»[5] – si proponevano di superare.

La possibilità di acquisire e trattare un numero vastissimo di informazioni in un contesto di contiguità, riproducendo il rapporto dialettico tra lo storico e i suoi documenti, si è concretizzata, dagli anni Novanta in poi, nella codifica digitale: è attraverso questo termine che passa la possibilità di riprodurre in un formato leggibile dal computer una fonte storica senza perdere quelle funzionalità di efficiente ricerca e di elaborazione dei dati consentite da

[4] Ulteriori difficoltà possono inoltre sorgere «e in modo più acuto, quando la base di dati costruita dallo storico A debba essere consultata per un'altra ricerca dallo storico B», R. ROWLAND, *Fonti, basi di dati e ricerca storica*, in *Storia & Computer. Alla ricerca del passato con l'informatica* cit., pp. 48-63: 54.

[5] R. BUSA, *Informatica e nuova filologia*, in *Lessicografia, filologia e critica*. Atti del Convegno Internazionale di Studi (Catania-Siracusa, 26-28 aprile 1985), a cura di G. SAVOCA, Firenze, L. S. Olschki 1985 (Biblioteca dell'Archivum romanicum, s. II, Linguistica, 42), pp. 17-25:19.

una gestione strutturata dell'informazione[6]. A livello zero, ogni testo informaticamente trascritto viene immediatamente codificato dalla macchina mediante una rappresentazione binaria (0 e 1) in formato ASCII[7] o UNICODE[8].



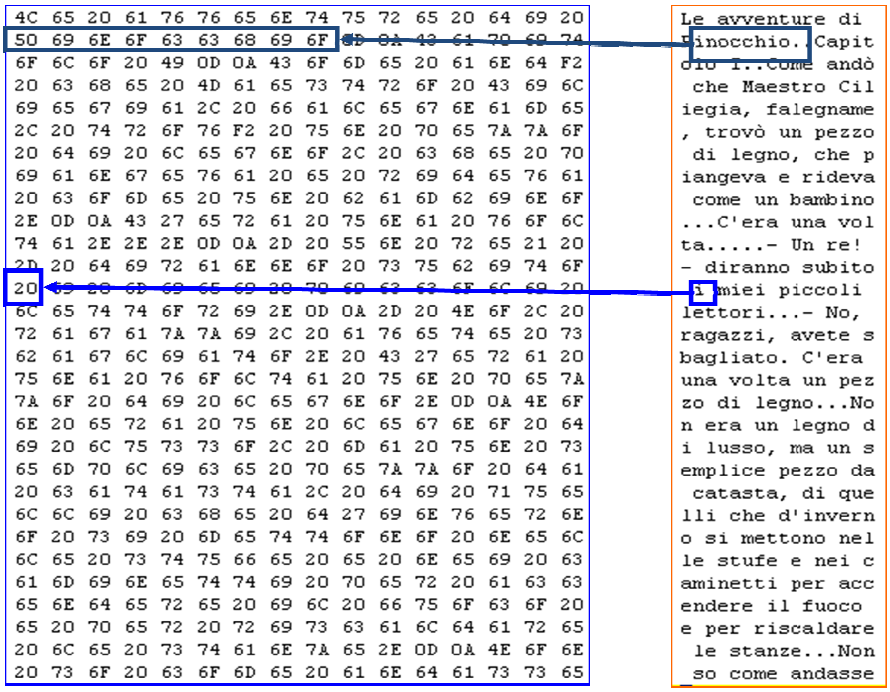
La codifica binaria di una lettera alfabetica e la tabella ASCII[9]

[6] «Definiamo come codifica un procedimento per mezzo del quale i dati che compongono un'informazione vengono materializzati e possono diventare un messaggio», G. GIGLIOZZI, *Introduzione all'uso del computer negli studi letterari*, a cura di F. CIOTTI, Milano, Bruno Mondadori 2003 (Campus), p. 21.

[7] Il codice ASCII, primo standard per l'assegnazione di codici a caratteri (1963), fornisce una tabella di corrispondenza che associa un numero ad ogni elemento di un insieme di 128 caratteri, comprendenti i principali caratteri dell'alfabeto latino, i principali segni di interpunzione e un certo numero di caratteri speciali. Nella rappresentazione decimale, i numeri associati ai caratteri sono compresi tra 0 e 127, nella rappresentazione binaria tra 0 e 1111111. La tabella così ottenuta permette di rappresentare ognuno dei caratteri codificati attraverso 7 bit di informazione, che conterranno la cifra binaria associata al carattere corrispondente: ad esempio, il numero decimale associato alla lettera maiuscola A è 65, e il corrispondente numero binario è 1000001. Nel tempo, l'ASCII stretto a 7 bit è stato sostituito dal codice ASCII esteso, a 8 bit, con il quale è possibile rappresentare 256 caratteri. L'estensione ASCII più importante è denominata ISO Latin 1. Per le specifiche del codice ASCII cfr. http://webopedia.internet.com/TERM/A/ASCII.html; per una descrizione del codice ISO Latin 1: http://www.hut.fi/u/jkorpela/latin1.

[8] La tavola UNICODE nasce per superare le limitazioni del codice ASCII: la sua codifica è basata su 16 bit, che consentono ben oltre 65.000 diverse combinazioni di 0 e 1; la versione 2.0 comprende attualmente 38.885 caratteri e rappresenta un sforzo immenso di informatizzazione non solo dal punto di vista informatico, ma anche da quello linguistico. Il sito ufficiale del progetto è http://www.unicode.org.

[9] La tabella è stata scaricata dal sito della *ISO* (*International Organization for Standardization*): http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=28245



Il "dietro le quinte" di un testo formattato[10]

Il testo inteso come sequenza di caratteri non coglie però che una piccola parte dell'informazione testuale, e le sue strutture profonde rimangono per lo più implicite e nascoste. Il prodotto di una codifica di basso livello è cioè un surrogato, per di più parziale, dell'opera originaria, in cui si ha completa equivalenza solo dal punto di vista dei caratteri che lo compongono e nessun guadagno di informazione. La codifica binaria dei caratteri non esaurisce i problemi di rappresentazione delle caratteristiche di un testo, che è oggetto complesso caratterizzato da molteplici livelli strutturali, non limitabili alla sequenza di simboli del sistema di scrittura: il dato codificato attraverso una semplice trasposizione binaria resta grezzo e non rappresenta una fonte esplicita di informazione.

Ad un livello superiore è invece possibile rappresentare il testo su supporto digitale in formato *Machine Readable Form* – utilizzabile dunque dall'elaboratore – razionalizzando le informazioni disponibili, classificandole e introducendovi attributi mediante un opportuno linguaggio teorico, il *markup language*, in grado di descrivere il documento in ogni sua parte attraverso l'apposizione di marche, ovvero stringhe di carattere delimitate da due parentesi uncinate. Queste marche sono dette, in termini informatici, *tag*: si tratta, in sostanza, di metadati con funzione identificativa, contenuti all'interno di un documento e riconoscibili dal processore informatico ai fini di un trattamento informatico. Una codifica

[10] L'immagine è tratta dalle slide del corso di Linguistica Computazionale "*Metodi computazionali per l'esplorazione e la rappresentazione dei dati linguistici*", tenuto da Alessandro Lenci (Dipartimento di Linguistica dell'Università di Pisa), disponibili on line all'indirizzo: www.humnet.unipi.it/dott_linggensac/materiale/corsointroduttivo-2005.ppt.

di alto livello cioè, è in grado di arricchire il testo formalizzato al livello zero con informazioni relative alle sue dimensioni strutturali, organizzandole in strutture macrotestuali e rendendo esplicita qualsiasi interpretazione, anche di tipo linguistico, si voglia associare al testo. Scopo di un linguaggio di codifica, i cui presupposti teorici sono ovviamente la teoria dell'informazione e della sua rappresentazione di Shannon[11], è dunque quello di identificare le strutture e le relazioni intercorrenti tra i dati testuali di un documento, scomponendoli in elementi discreti e assegnando una struttura alla rappresentazione in grado di distinguere, nella sequenza di caratteri codificati, parti diverse con funzioni diverse, creando per questa via il presupposto per un corretto funzionamento degli strumenti di gestione e ricerca automatica sul *corpus* testuale. Attraverso un linguaggio di marcatura l'informazione è scomponibile in dimensioni realmente minime: ad ogni livello, dal più elevato – ad esempio l'intero documento – al minore – il paragrafo, la frase, la parola, la singola lettera – è infatti possibile riconoscere e assegnare un valore semantico. In questo senso anche il termine codice assume un significato diverso e, forse, più ampio: non solo strumento per trasferire informazioni da un sistema all'altro, da una lingua all'altra, ma complesso meccanismo che modella la (e si modella sulla) materia trattata[12].

Nell'affrontare le specifiche dei linguaggi di marcatura è però opportuno fornire alcuni chiarimenti, partendo dalla definizione canonica fornita da Gerard Genette, che per codifica ha inteso:

---

[11] Claude Shannon, che nel suo *La teoria matematica della comunicazione* ha proposto di utilizzare il concetto di scelta (o decisione) per misurare la quantità di informazione contenuta in un messaggio, è il padre della moderna teoria dell'informazione. Lavorando all'*information theory*, l'obiettivo di Shannon era solo quello di eliminare i disturbi dai collegamenti telefonici; ma la teoria dell'informazione cui approdò rappresenta una delle più importanti conquiste teoriche del XX secolo, e ha avuto delle profonde ricadute nel campo delle applicazioni telematiche. Nel suo lavoro, Shannon si è interrogato su quali aspetti distinguere all'interno di un processo comunicativo, osservando come una distinzione tra la sfera tecnica della comunicazione e quella relativa ai suoi contenuti semantici possa portare ad un miglioramento nella comprensione delle caratteristiche del processo, cfr. C.E. SHANNON, W.WEAVER, *The mathematical theory of communication*, Urbana, University of Illinois press 1949.

[12] Cfr. G. GIGLIOZZI, *Codice, testo, interpretazione*, in *Studi di codifica e trattamento automatico di testi* cit., pp. 65-84:66. Il concetto di codice rappresenta una delle nozioni chiave di ogni disciplina che si occupa di processi comunicativi, ma il suo significato non è sempre univoco. Una definizione generale su cui convenire è la seguente: «un codice è un insieme strutturato di segni e regole che il mittente e il destinatario devono condividere affinchè il primo sia in grado di formulare dei messaggi e il secondo di comprenderli», F. CIOTTI, G. RONCAGLIA, *Il mondo digitale. Introduzione ai nuovi media*, Roma-Bari, Laterza 2000 (I Robinson Letture), p. 288. In questo senso la nozione di codice è coestensiva a quella di linguaggio. Nell'ambito trattato in questa sede, il termine codice va inteso in accezione semiotica, come sistema di correlazione arbitrario tra due sottosistemi che costituiscono, alternativamente, il sistema delle unità significanti che si manifestano in un atto comunicativo (piano dell'espressione) e il sistema delle unità significate (piano del contenuto). La forma dell'espressione è la struttura che organizza e dà forma alle unità significanti, fornendo un repertorio di tipi espressivi del codice e le regole per la loro combinazione; la forma del contenuto invece, definisce le unità semantiche e i loro rapporti, organizzando la conoscenza/rappresentazione del mondo in un sistema.

> la rappresentazione di un testo attraverso un linguaggio formale in grado non solo di dare istruzioni di superficie sull'aspetto del testo, ma anche di costruire l'identità del documento attraverso la sua fruizione. Il contenuto e l'organizzazione delle etichette (metadati) guida l'acceso alla risorsa, è una sorta di gioco di specchi: creo una risorsa e mentre la trascrivo ne costruisco l'accesso[13].

Il nodo centrale dell'enunciazione di Genette sembra essere, ovviamente, quello relativo all'introduzione dei metadati nella costruzione del testo codificato[14]. La letteratura in materia ha cercato, in modo finora insoddisfacente, ripetitivo e inutilmente sovrabbondante, di definirne i confini, la natura, le funzioni: di particolare rilevanza – ma non di altrettanta efficacia – ad esempio, lo sforzo prodigato in questi anni nel campo da parte di alcune comunità di pratiche, dai ricercatori in campo scientifico ai bibliotecari, che hanno condotto a un'ipotesi di classificazione non particolarmente felice ma ormai largamente utilizzata e tradotta nelle norme NISO 2004. Di fatto, negli sviluppi implementativi, i metadati per la conservazione come informazioni necessarie per archiviare e conservare una risorsa al fine di assicurarne l'autenticità e la possibilità di riproduzione e ricostituzione, si limitano a identificare e gestire informazioni di natura quasi esclusivamente tecnologica[15] e sono comunque difficilmente riferibili a documenti digitali complessi. Semplificando, si potrebbero comunque intendere i metadati come:

- dati che forniscono informazioni su una fonte informativa;
- informazioni che caratterizzano i dati;

[13] G. GENETTE, *Soglie. I dintorni del testo*, Torino, Einaudi 1989 (Einaudi Paperbacks, 195), p. 5.

[14] Metadati e dati si definiscono in relazione l'uno con l'altro: i primi vengono considerati tali solo in seguito ad una scelta, e non lo sono per natura. La distanza tra dato e metadato non è dunque separata da una scelta alternativa che porrebbe i due oggetti in termini dicotomici: piuttosto, vi è una scala graduata, un *continuum* che lascia intravedere delle zone grigie in cui i dati tendono a confondersi con i metadati. Il concetto proviene dalla teoria delle basi di dati, cioè dall'organizzazione di sistemi di informazioni strutturate di rilevanza amministrativa e tecnica di cui i metadati identificano – tra l'altro – la struttura, la natura, la fonte e ne consentono l'accesso e l'utilizzo. Dunque, sostanzialmente, con il termine *metadati* si indica l'insieme di dati e informazioni che descrivono una risorsa o un documento digitale, divisi nelle tre classi di metadati descrittivi, metadati gestionali o amministrativi, metadati strutturali. «Si tratta di informazioni che nei sistemi documentari tradizionali sono espresse in modo quasi sempre esplicito nel documento stesso e solo in casi assai limitati costituiscono il risultato di procedure esterne al sistema», ma che svolgono una funzione cruciale per la creazione di liste e indici, cfr. M. GUERCIO, *Archivistica Informatica* cit., p. 34. I formati di metadati comunemente usati in ambito archivistico e bibliotecario sono: Dublin Core (http://dublincore.org/), METS (http://www.loc.gov/standards/mets), MODS (http://www.loc.gov/standards/mods/), MIX (http://www.loc.gov/standards/mix/). Sull'argomento v. anche P. HORSMAN, *Metadata: concetto archivistico o territorio informatico*, in *La conservazione dei documenti informatici. Aspetti organizzativi e tecnici* (Roma, 31 ottobre 2000), in *Archivi & Computer*, 1 (2001), pp. 35-43 e G. MICHETTI, *Standard e metadati: concetti nuovi per l'archivistica?*, in *Nuovi annali della Scuola speciale per archivisti e bibliotecari*, 14 (2000), pp. 229-253.

[15] «*Information that supports and documents the process of digital preservation: the term is usually reserved for metadata that specifically supports the functions of maintaining the fixity, viability, renderability, understandability, and/or authenticity of a digital material in a preservation context*», P. CAPLAN, *Preservation metadata. Report for DCC*, London 2006, p. 134.

- dati utili ad identificare caratteristiche condivise da più documenti;
- dati strutturati su dati.

Incorporati all'interno del testo e denominati alternativamente codifica (*encoding*), marcatura (*markup*) e, con un brutto calco, taggatura (*tagging*), i metadati permettono di assegnare una struttura alla rappresentazione testuale distinguendo, nella sequenza dei caratteri codificati, parti diverse con funzioni diverse. Si tratta, a ben vedere, di elementi metalinguistici che, interni al documento, raffigurano in qualche modo un'estensione dello stesso sistema descritto, un ampliamento delle risorse espressive del testo in funzione autoriflessiva, permettendo di esplicitare quegli elementi che altrimenti vi resterebbero impliciti. I nodi concettuali di questa operazione sono allora:

- la possibilità di identificare le strutture e le relazioni che intercorrono tra i diversi elementi di un documento;
- l'obbligo di effettuare un'analisi degli elementi del testo e del suo contesto;
- il suo configurarsi, contemporaneamente, come parte del testo e informazione sul testo;
- la capacità di mettere in chiaro, rendere manifesti ed evidenziare i vincoli strutturali sottesi alla fonte in esame.

Ma, ed è bene sottolinearlo, i termini stessi con cui la codifica viene realizzata tradiscono un'origine niente affatto innovativa. Il *markup* è concetto derivato dal gergo tipografico inglese, con cui ci si riferiva alle annotazioni che un editore apponeva in margine al testo per assistere il compositore nell'impaginazione del testo a stampa[16], ed è presente a diversi livelli in ogni forma testuale: si pensi alla consuetudine, nella scrittura geroglifica egiziana, di evidenziare i nomi personali con un ovale o di colorare le frasi significative, o ai più comuni e diffusi marcatori dei moderni sistemi alfabetici, relativi alla spaziatura tra le parole, la punteggiatura, i segni diacritici, l'alternanza di lettere maiuscole e minuscole. Rispetto alle convenzioni della scrittura però, il *markup* informatico si configura, più propriamente, come il frammento nascosto del linguaggio dell'oggetto, il metalinguaggio che lo descrive, la trascrizione diplomatica ad uso del computer[17]. In questo senso è

[16] Cfr. E. PIERAZZO, *La codifica dei testi. Un'introduzione*, Roma, Carocci 2002 (Beni Culturali, 29).
[17] Cfr. D. BUZZETTI, *Archiviazione digitale dei dati e adeguatezza della rappresentazione del testo*, in *Schede Umanistiche*, 9 (1999) 2, pp. 209-218:214.

contemporaneamente parte del testo che dice qualcosa sul testo, un approccio strutturato che si pone tra il documento come fonte e la sua visualizzazione, consentendo la sua memorizzazione secondo logiche di formalizzazione assai più flessibili e sofisticate dei tradizionali sistemi di *database*.

Nell'ambito del *markup,* due sono le gradi categorie di riferimento:

1. i linguaggi di marcatura dichiarativa (logica o descrittiva), in cui i marcatori indicano la funzione assolta dal blocco di testo a cui si riferiscono, dichiarando la sua appartenenza ad una determinata classe di strutture;
2. i linguaggi di marcatura procedurale (o tipografica), che consistono in una serie di istruzioni operative indirizzate alla formattazione e all'impaginazione del testo, inserendo metadati di carattere tipografico che forniscono istruzioni al *software* per la produzione di un *output* del documento.

Se si fa riferimento ad un programma, la prima idea è quella di fornire al computer, una dopo l'altra, una serie di istruzioni da eseguire, una serie di ordini dati sequenzialmente alla macchina che li esegue: una procedura insomma[18]. La marcatura procedurale, dipendente dal sistema, associa infatti ad ogni elemento del documento il procedimento per visualizzarlo nella maniera voluta (carattere, dimensione, corsivi, grassetti, margini, interlinea). Al contrario, un linguaggio dichiarativo non fornisce ordini al calcolatore, ma glieli spiega, riferendo qual è il problema da risolvere, quali sono le caratteristiche della situazione, quali sono gli elementi coinvolti e come possono essere modificati. Un linguaggio dichiarativo cioè, descrive, dichiara i dati del problema: al computer è demandato il compito di analizzarlo e dedurre la risposta da dare. Per assolvere questo compito, il *markup* dichiarativo si fonda sul ruolo di ogni elemento all'interno del testo e in questo senso è indipendente dal sistema ma contestuale, perché in grado di specificare le regole di correttezza dei documenti codificati.

Il progenitore dei linguaggi di marcatura dichiarativa è *SGML* (*Standard Generalized Markup Language*), un metasistema di codifica nato con lo scopo di stabilire i costrutti sintattici e semantici di un linguaggio di *markup* finalizzato alla creazione, manipolazione e gestione di documenti elettronici non legati ad una determinata architettura *hardware* o *software.* Ideato da Charles Goldfarb nel 1974 e consolidatosi dalla metà degli anni Ottanta,

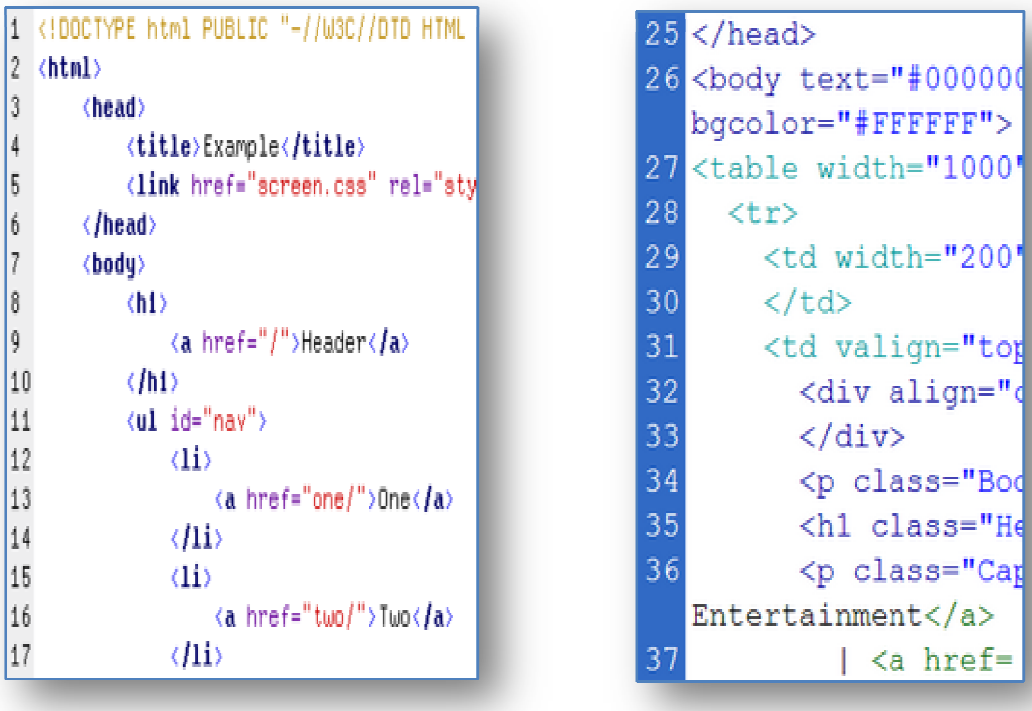
[18] Tra i linguaggi procedurali più famosi, vanno citati Pascal, Fortran, Basic, C. Sull'argomento v. C. GHEZZI, M. JAZAYERI, *Programming language concepts*, New York, J. Wiley 1987.

SGML rappresenta il risultato di oltre vent'anni di sforzi profusi per la standardizzazione di un meccanismo generale di definizione di stili di marcatura diversi, traendo origine dallo *GML* (*Generalized Markup Language*) già avviato nel 1969 nei laboratori IBM con lo scopo di supportare l'elaborazione informatica di documentazione legale, tecnica e amministrativa. Il linguaggio elaborato dall'IBM, introduceva il concetto di "tipo di documento" come classe con precise regole di struttura e formattazione, definibili attraverso uno schema di marcatura. Goldfarb vi aggiunse un sistema di collegamento tramite riferimenti semantici e l'idea di unificare gli ordini di impaginazione di un testo superando l'inconveniente dell'esistenza di molteplici linguaggi, ciascuno con una sua sintassi, legati ai diversi programmi di impaginazione automatica in uso (le famiglie dei *troff, LaText*) [19]. Secondo le sue direttive, *SGML* era rivolto agli editori e agli organi amministrativi, venendo incontro all'esigenza di conservare le informazioni contenute non nel testo in sé ma nella sua disposizione, e la possibilità di supportare lo scambio e la trasmissione di documenti tra enti e gruppi senza perdita di informazioni rilevanti. Ma è stato a partire dalla creazione di *SGML* che il *markup* descrittivo ha assunto un notevole interesse anche per la comunità scientifica, offrendo una base per affrontare efficacemente i problemi di rappresentazione informatica del materiale testuale e documentario attraverso la definizione di raccomandazioni per la creazione dei testi in *Machine Readable Form.*

La complessità della struttura sintattica proposta da *SGML* ne hanno resa ardua un'effettiva implementazione. Lo standard ha però costituito la base sintattica attraverso cui, alla fine degli anni Ottanta, Tim Berners-Lee[20] ha sviluppato l'*HTML* (*Hyper Text Markup Language*) che dal 1991 fonda la struttura portante del sistema internet, il World Wide Web. Sfruttando il concetto di *tag*, ogni elemento da visualizzare nella pagina in linguaggio *HTML* è infatti rappresentato da una struttura comprendente un'etichetta iniziale, al cui interno sono inseriti nomi e attributi, seguita da un ulteriore contenuto e da un marcatore finale.

[19] Per la storia e le specifiche tecniche di *SGML* cfr. C.F. GOLDFARB, *The SGML handbook*, Oxford 1991 e il sito: http://www.w3.org/MarkUp/SGML.

[20] Tim Berners-Lee è il co-inventore del World Wide Web insieme a Robert Caillau, realizzato nel 1980 presso il CERN di Ginevra come programma (inizialmente chiamato *Enquire*), ad uso privato, per immagazzinare informazioni usando associazioni casuali. Sulla nascita del web e del linguaggio *HTML* cfr. T. BERNERS-LEE, *L'architettura del nuovo Web*, Milano, Feltrinelli 2001 (Interzone); l'*Home Page* di Berners-Lee è raggiungibile all'indirizzo: http://www.w3.org/People/Berners-Lee/.



Due esempi di codice *HTML* con sintassi evidenziata[21]

La semplicità della tecnologia proposta da Berners-Lee, sebbene inizialmente gli standard e i protocolli abbiano supportato esclusivamente la gestione di pagine .html statiche, ha avuto un grande successo, sia in campo accademico che in quello commerciale, dando inizio a quella che oggi viene chiamata l'era del Web[22]. Ma come linguaggio di marcatura finalizzato a supportare l'editoria digitale *HTML* ha mostrato fin da subito evidenti segni di debolezza, a causa del tipo di codifica implementato, di natura procedurale piuttosto che dichiarativa, in cui le istruzioni di marcatura sono tipografiche e stilistiche, limitandosi a segnalare all'*editor* dove e come i testi e i loro segmenti, le immagini, i collegamenti, debbano disporsi sulla pagina elettronica. I limiti principali dell'*HTML* nel campo della ricerca storica sono inoltre legati alla strutturale incapacità di questo linguaggio di fornire un'adeguata rappresentazione dell'informazione, alla sua immodificabilità e chiusura, alla scarsa articolazione interna e, in ultima analisi, ad una sintassi poco potente, incapace di descrivere fenomeni testuali complessi. Dagli ostacoli di natura rappresentazionale sono

[21] L'immagine è tratta da Wikipedia: http://it.wikipedia.org/wiki/HTML.

[22] Negli ultimi anni il linguaggio *HTML* ha subito numerose revisioni e miglioramenti, passando dalla versione 1.0 fino a 3.2 e arrivando alla versione 4.0 e 4.01 (per le cui specifiche si rimanda all'indirizzo: http://www.w3.org/TR/html4/). La versione HTML 3.2, utilizzata dai cosiddetti *browser* di terza generazione, permetteva di regolare gli allineamenti delle celle della tabella al punto, migliorando così, rispetto alla precedente versione, il lavoro dei *designer*; la versione 4.0 ha permesso di separare contenitore da contenuto, aggiungere supporto per nuove tecnologie, migliorare l'accesso web ai portatori di handicap, cfr. http://www.w3.org/MarkUp/HTML.

derivati, conseguentemente, forti limiti operativi: la ristretta consistenza strutturale ha infatti ostacolato la creazione automatica e dinamica di indici e sommari, costringendo ad esempio i motori di ricerca a riferire come esito un documento intero (la pagina .html) e non l'informazione richiesta, rendendo dunque difficoltoso e poco significativo il *retrieval*.