

VERSO L'EDIZIONE CRITICA IPERTESTUALE
DELLO ZIBALDONE LAURENZIANO (PL. XXIX. 8) AUTOGRAFO
DEL BOCCACCIO. STATO D'AVANZAMENTO DEL PROGETTO.

1. *Descrizione dell'esperienza*

Il progetto di un'edizione critica elettronica del manoscritto Laurenziano autografo del Boccaccio è un'iniziativa progettata dal prof. Raul Mordenti del Dipartimento di Studi filologici, linguistici e letterari (Università "Tor Vergata", Roma) che si avvale della collaborazione del Centro Interdipartimentale Servizi per l'Automazione delle Discipline Umanistiche (Università "La Sapienza", Roma), nonché della Dott.ssa Alda Spotti della Biblioteca Nazionale Centrale di Roma "Vittorio Emanuele II" per quanto riguarda la trascrizione.

Il manoscritto, conservato presso la Biblioteca Medicea Laurenziana di Firenze, contiene come è noto una raccolta di testi trascritti e/o composti dal Boccaccio. Esso pone notevoli problemi sia di natura filologica sia di natura critica, tra i quali la questione della definitiva attribuzione al Boccaccio anche delle prime 45 carte del manoscritto. Mancano tuttavia edizioni integrali dell'opera, come degli altri Zibaldoni del Boccaccio, che pure sarebbero un utile strumento di studio e di approfondimento.

Scopo del progetto è da una parte quello di fornire un'edizione elettronica del manoscritto comparabile ad una edizione critica per metodo e precisione, dall'altra di inserire tale edizione in un sistema ipertestuale in cui l'utente possa disporre delle edizioni finora pubblicate, seppure riferite a singoli segmenti testuali, delle immagini digitali delle carte del manoscritto Laurenziano, di una bibliografia di supporto.

Il modello dell'ipertesto è stato presentato al convegno *Gli Zibaldoni di Boccaccio: memoria, scrittura e riscrittura*, svoltosi con il patrocinio dell'Ente Nazionale 'Giovanni Boccaccio' a Firenze e Certaldo dal 26 al 28 aprile 1996 (cfr. R. Mordenti, *Problemi e prospettive di un'edizione ipertestuale dello Zibaldone Laurenziano*, nel volume degli Atti del convegno in corso di stampa presso l'Editore fiorentino Franco Cesati); tale presentazione ha segnato la conclusione della prima fase progettuale consistente nella definizione di un sistema aperto in cui poter far confluire tutti i dati disponibili sullo Zibaldone Laurenziano Pl. XXIX. 8 e sui segmenti testuali in esso contenuti.

Nella fase successiva le nostre energie sono state dedicate allo studio di un modello di codifica in grado di descrivere la struttura del manoscritto e tutte quelle caratteristiche in esso presenti ritenute significative per un'analisi e una valutazione globale dell'opera.

La trascrizione in formato elettronico non viene intesa in questa sede semplicemente come la trasposizione di un testo su un supporto diverso da quello della pergamena, cioè come una mera trascrizione o riproduzione dell'originale di tipo 'imitativo'. Si intende piuttosto produrre un documento elettronico arricchito di informazioni relative ai fenomeni paleografici e codicologici e filologici e linguistici considerati rilevanti e comunque suscettibili di essere utilizzati per la risoluzione di alcuni dei problemi del nostro testo ancora aperti alla discussione; in altre parole si intende utilizzare le potenzialità euristiche consentite dall'informatica, ma per far questo è necessario concentrare il massimo dell'attenzione sul momento, assolutamente cruciale, della codifica.

Per codificare lo Zibaldone abbiamo scelto di utilizzare lo Standard Generalized Markup Language (SGML), pubblicato dall'International Standard Organization con il titolo "Information Processing — Text and Office System — Standard Generalized Markup Language (SGML)", n. 8879, 1986. Si tratta di un metalinguaggio che fornisce una sintassi e delle regole per descrivere la struttura di un documento. Caratteristica fondamentale della codifica SGML è proprio l'essere rivolta alla descrizione della struttura interna di un documento e delle sue componenti, non al modo di presentazione a video o a stampa. Ne consegue che il documento SGML si configura come una sorta di archivio al cui interno sono automaticamente riconoscibili nuclei di informazione. Va poi sottolineato che SGML è un 'linguaggio di codifica' e non un 'sistema di codifica', in altre parole è un sistema che permette di definire di volta in volta il proprio modello descrittivo senza imporre gabbie che risulterebbero inevitabilmente poco idonee in un settore come quello delle discipline umanistiche in cui i documenti da esaminare non sono riconducibili a categorie rigide ma presentano un'infinita variabilità di forme.

Il modello di struttura elaborato, definito in SGML come Document Type Definition (DTD), è stato da noi sperimentato codificando alcune parti del manoscritto ed è stato presentato alla giornata internazionale di studio *Boccaccio e Chaucer* tenutasi a Certaldo il 21 giugno 1997. Abbiamo ritenuto opportuno produrre una DTD specifica e non utilizzare quella della Text Encoding Initiative (TEI) perché, come vedremo meglio nel paragrafo successivo, la gerarchia da noi scelta per rappresentare il testo riflette la struttura fisica *del manoscritto*, mentre la TEI-DTD per sua natura è basata sulle divisioni interne *del testo* da codificare. Per adattare la TEI-DTD all'altra gerarchia sarebbe stato necessario modificarla in modo piuttosto radicale e, sebbene tale possibilità sia prevista dagli autori, si è deciso di ricorrere alla sintassi SGML e produrre una DTD autonoma (appositamente elaborata da chi scrive) rispetto alla TEI.

2. Alcuni concetti fondamentali dello SGML e la DTD per lo Zibaldone

SGML è un linguaggio da utilizzare per descrivere dei testi mediante l'uso di etichette/tag; concetto fondamentale è il riconoscimento di una struttura gerarchica che forma il documento, costituita da oggetti contenenti informazioni, ordinati gerarchicamente. Queste categorie coerenti al loro interno per il tipo di contenuto informativo, sono chiamate nella sintassi SGML 'elementi'. La citazione ad esempio può essere un elemento di una DTD; tutte le volte che nel documento si incontra una citazione essa costituisce un'occorrenza della classe 'citazione' e viene delimitata con l'inserimento di un tag di apertura e di uno di chiusura.

Il modello di codifica (cfr. Fig. 1 e Tab. 1 a p. 118) da noi applicato all'edizione dello Zibaldone non è molto complesso. La gerarchia essenziale è quella data dalla successione delle carte nel manoscritto, ciascuna carta è composta da due pagine, cioè da un *recto* e da un *verso*; ogni pagina comprende una o più colonne di testo. La colonna può presentare, oppure no, al suo interno, dei raggruppamenti di righe (ad esempio delle strofe). L'unità minima della gerarchia è rappresentata dal rigo.

Gli elementi SGML possono avere uno o più attributi, introdotti nella DTD dalla parola chiave ATTLIST. Si tratta di informazioni supplementari riferite all'elemento che vengono inserite nel codice di apertura (open-tag) di ogni occorrenza del medesimo elemento. Nel caso presente ogni elemento <carta> avrà come attributo il numero di successione nel manoscritto; ogni elemento <pag> la qualifica *recto* o *verso*; ogni elemento <col[onna]> un codice convenzionale che identifica colonne a piena pagina (a), prima colonna di due o tre (b), seconda colonna (c), etc. Infine il <rigo> avrà più attributi che definiscono rispettivamente il numero di successione nella colonna, se esso è rientrato rispetto al margine sinistro della colonna, oppure no, ed in caso affermativo, di quanti spazi-lettera è tale rientro. Si intuisce facilmente che gli attributi diventano uno strumento fondamentale per le successive analisi in quanto consentono di introdurre direttamente nel testo osservazioni di varia natura in un formato standard gestibile dalla macchina.

L'elemento <note> viene utilizzato per codificare le glosse assai numerose e significative nel manoscritto. Esse vengono inserite alla fine di ogni pagina del testo, mentre gli attributi specificano in ogni caso la posizione, l'orientamento e la mano che ha tracciato il testo della glossa (per indicare la posizione si è convenzionalmente divisa la pagina in quadranti usando i punti cardinali).

Alla gerarchia fondamentale descritta sono stati aggiunti una serie di elementi dei quali si sta tuttora verificando la completezza, che descrivono anche mediante gli attributi caratteristiche codicologiche e paleografiche. Si tratta tuttavia di elementi che non sono legati alla gerarchia proposta, che riflette la struttura del supporto pergameneo, e che pertanto sono stati definiti come elementi 'inclusive' rispetto all'elemento radice della DTD. Nella sintassi SGML gli 'inclusive'

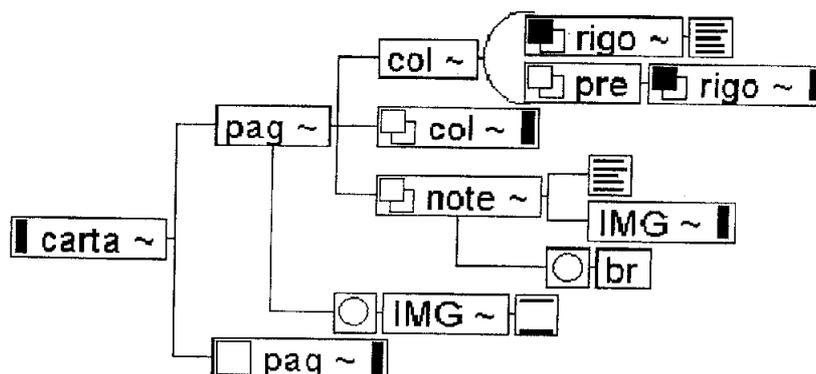


Fig. 1. Rappresentazione grafica della struttura dell'elemento <carta>.

<!ELEMENT	carta	- O (pag,(pag?)) >	
<!ATTLIST	carta		
	n	CDATA	#IMPLIED >
<!ELEMENT	pag	- O (col,(col*), (note*))>	
<!ATTLIST	pag		
	n	(r,v)	#IMPLIED >
<!ELEMENT	col	- O (rigo pre)+ >	
<!ATTLIST	col		
	type	(a,b,c,d,e,p)	#IMPLIED >
<!ELEMENT	pre	-- (rigo+) >	
<!ELEMENT	rigo	- O (#PCDATA) >	
<!ATTLIST	rigo		
	n	NUTOKEN	#IMPLIED
	rientrato	(silno)	no
	rientro	NUMBER	#IMPLIED>
<!ELEMENT	note	-- ((#PCDATA img)+ +br>	
<!ATTLIST	note	ID	#IMPLIED
	n	CDATA	#IMPLIED
	resp	CDATA	#IMPLIED
	place	(nolnelenles! onlosolselse lolnls)	#IMPLIED
	place	CDATA	#IMPLIED
	orient	(oriz vert obliva oblivb)	#IMPLIED
<!ELEMENT	br	- O EMPTY	

Tab. 1. La tabella 1 presenta le righe di definizione degli elementi della DTD secondo le regole SGML, equivalenti alla struttura presentata in Fig. 1. in formato grafico.

elements' sono elementi che possono ricorrere in qualunque posizione all'interno dell'elemento al quale sono collegati.

Le caratteristiche che vengono codificate secondo la DTD predisposta (cfr. Fig. 2 e Tab. 2 alle pp. 120-121) sono le seguenti:

Abbreviazioni: per ciascuna abbreviazione si trascrive lo scioglimento codificandolo con il tag <abbrev>. Gli attributi definiti servono per dichiarare il tipo (tipo) di abbreviazione ed il glifo utilizzato per rappresentarla (rend) utilizzando una serie di codici numerici.

Caratteri: il tag viene utilizzato per codificare caratteri che presentano anomalie. In particolare: caratteri scritti con inchiostro colorato, o di dimensioni diverse da quelle medie del testo, o non scritti sul rigo di base, o infine appartenenti ad alfabeti diversi dal latino. Gli attributi specificano dunque il colore, le dimensioni, la posizione, l'alfabeto di riferimento.

Caratteri o stringhe su cui è stata effettuata una correzione: l'elemento <correz> ha un contenuto complesso, può contenere infatti altri elementi che individuano caratteri cancellati () o aggiunti in un secondo momento (<add>). Caratteri cancellati/aggiunti: gli attributi del tag indicano il tipo di cancellature la responsabilità, se nota, cioè quale mano ha effettuato la correzione; inoltre se il testo cancellato è leggibile o meno. Nel caso di caratteri aggiunti viene specificata ancora la mano, se nota, e la posizione rispetto al rigo di base.

Iniziali decorate: un tag specifico per le iniziali, <iniziale.decorata>, consente di specificare anche l'altezza dell'iniziale (il numero di righe corrispondenti), se è miniata o meno, la posizione rispetto al margine della colonna e la eventuale presenza di una lettera di richiamo inserita dal copista.

Spazi: gli spazi vuoti lasciati dal copista sono codificati con il tag <spazio>.

Interventi editori: il tag <aggiunte> delimita aggiunte editoriali.

Testo illeggibile: il testo illeggibile per cause materiali viene codificato come <unclear>. Gli attributi specificano il numero dei caratteri illeggibili, l'integrazione proposta, la causa materiale che rende impossibile la lettura.

Testo indecifrabile: il testo che gli editori non riescono a leggere viene codificato come <indecif>. In questi ultimi due elementi si può anche inserire l'integrazione nella trascrizione codificandola come <hp> o congettura.

Ultimo elemento presente nello schema è <index>, da utilizzare per indicizzare in un secondo momento il testo; negli attributi viene dichiarato il tipo di indice nel quale inserire la voce e la forma normalizzata.

Un'ultima osservazione riguarda i codici numerici utilizzati per memorizzare il testo. I segni alfabetici del manoscritto corrispondenti ai nostri e appartenenti alla competenza del copista vengono codificati utilizzando il set ASCII non esteso (127 codici numerici); per quei segni invece che non hanno corrispondenza con l'attuale set alfabetico, ad esempio il segno di paragrafo, sono state definite delle 'entità'; SGML consente di introdurre nel documento delle stringhe di

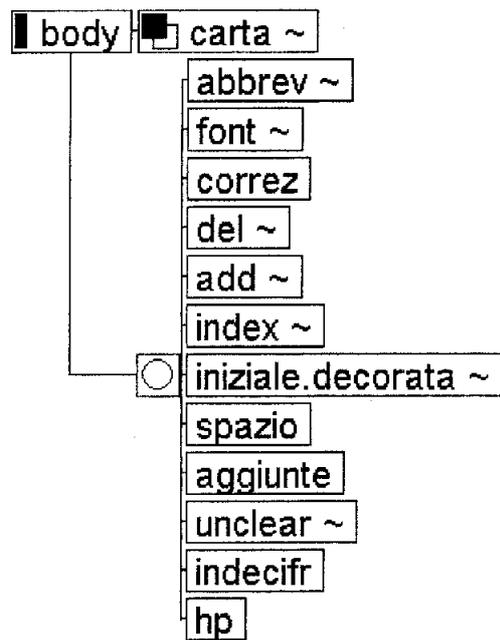


Fig. 2. Rappresentazione grafica della struttura dell'elemento <body>.

<!ELEMENT	abbrev	-- CDATA>	
<!ATTILIST	abbrev		#IMPLIED
	rend	CDATA	#IMPLIED>
	tipo	(troncamento, contrazione, tachigrafia, acronimo)	
<!ELEMENT	font	-- (#PCDATA) >	
<!ATTILIST	font		#IMPLIED
	misure	CDATA	"inlinea"
	posizione	(inlincalsupra/sub)	"latino"
	tipo	(latino greco ebraico altro)	
	colore	(rubr azz nero seppia altri)	"nero">
<!ELEMENT	correz	-- ((del unclear), (add)?) - (correz) >	
<!ELEMENT	del	- O ((#PCDATA hp)*) >	
<!ATTILIST	del		#IMPLIED
	tipo	(erasaldepenlespunta)	#IMPLIED
	resp	CDATA	#IMPLIED >
	stato	(illeggibile/leggibile)	
<!ELEMENT	add	-- (#PCDATA) >	
<!ATTILIST	add		#IMPLIED
	resp	CDATA	#IMPLIED >
	posizione	(inlinea,supra,sub)	
<!ELEMENT	iniziale.	-- (#PCDATA) >	
<!ATTILIST	decorata		#IMPLIED
	decorata		#IMPLIED
	misure	CDATA	#IMPLIED
	posizione	(dentro fuori debordante)	#IMPLIED
	figure	(si no)	#IMPLIED
	richiamo	CDATA	#IMPLIED>
<!ELEMENT	index	- O (#PCDATA) >	
<!ATTILIST	index		#IMPLIED
	index	CDATA	#REQUIRED >
	lemma	CDATA	
<!ELEMENT	spazio	-- (#PCDATA) >	
<!ELEMENT	aggiunte	-- (#PCDATA) >	
<!ELEMENT	unclear	- O ((#PCDATA hp)*) >	
<!ATTILIST	unclear		#IMPLIED
	nrlettere	CDATA	#IMPLIED
	ipotesi	CDATA	#IMPLIED
	causa	CDATA	#IMPLIED >
<!ELEMENT	indecifr	- O ((#PCDATA hp)*) >	
<!ELEMENT	hp	-- (#PCDATA) >	

Tab. 2. La tabella 2 presenta le righe di definizione degli elementi della DTD secondo le regole SGML, equivalenti alla struttura presentata in Fig. 2. in formato grafico.

caratteri delimitate dai due segni speciali "&" e ":", chiamate 'entities', e di associare fuori del documento queste stringhe a del testo o a delle immagini in modo che il software possa eseguire automaticamente la sostituzione, se possibile. Le entità vengono utilizzate soprattutto per introdurre segni alfabetici speciali non compresi nel set ASCII standard in modo che il file SGML possa essere comunque utilizzato anche se non sono disponibili i font con la rappresentazione.

3. Riflessioni conclusive

La codifica SGML è stata definita tenendo conto della necessità di esplicitare fenomeni importanti alla comprensione del testo e di spiegarne il significato. I segni topografici corrispondono agli elementi della gerarchia che rappresenta la struttura del manoscritto. I segni alfabetici sono codificati utilizzando il set ASCII non esteso e le entità. Le caratteristiche grafiche significative vengono codificate come elementi non legati alla gerarchia; gli attributi ne definiscono il significato testuale.

Il modello presentato viene attualmente utilizzato per la trascrizione in formato elettronico dei diversi segmenti testuali dello Zibaldone; se necessario potranno essere operate correzioni e integrazioni della Document Type Definition anche nel corso del prosieguo del lavoro, ma si è già verificata la possibilità di eseguire alcune analisi automatiche sui campioni di testo codificato. Grazie alla codifica e ad alcuni semplici programmi scritti nella shell di Unix è possibile, ad esempio, estrarre e presentare in tabella i caratteri scritti in posizione soprilineare e calcolare la frequenza del fenomeno per ogni carattere; o ancora estrarre le abbreviazioni, dividerle per tipologia, individuare quali sono i glifi più comuni e quelli eccezionali.

Il lavoro di trascrizione, già di per sé operazione lunga e complessa, procede ora più lentamente per la necessità di codificare contestualmente i vari fenomeni registrati. Si ha però la convinzione che il testo elettronico finale costituirà un importante strumento di studio e che contribuirà alla risoluzione di alcuni dei problemi filologici e interpretativi posti dal manoscritto.

Ilaria Bonincontro