

Il computer come calcolatore – 4

Storia dell'Informatica e della Comunicazione Digitale

Federico Gobbo

federico.gobbo@uninsubria.it

Università dell'Insubria, Varese

© Alcuni diritti riservati.

A.A. 2009-10

La Biblioteca di Babele

I suoi scaffali registrano tutte le possibili combinazioni dei venticinque simboli ortografici (numero, anche se vastissimo, non infinito) cioè tutto ciò ch'è dato esprimere, in tutte le lingue. Tutto: la storia minuziosa dell'avvenire, le autobiografie degli arcangeli, il catalogo fedele della Biblioteca, migliaia e migliaia di cataloghi falsi, la dimostrazione della falsità di questi cataloghi, l'evangelo gnostico di Basilide, il commento di questo evangelo, il commento del commento di questo evangelo, il resoconto veridico della tua morte. Jorge Luis Borges (1974:63–64)

E naturalmente anche il testo di questa lezione. Vediamo qui cosa succede alla scrittura quando incontra il calcolatore.

Mediamorfosi e rimediazione

La distinzione tra “vecchie” (analogiche) e “nuove” (digitali) tecnologie tende a sparire: i media vengono piú e piú **domesticati**, i.e. addomesticati e domestici insieme, e si adattano continuamente per inserirsi nel nostro quotidiano, in un processo continuo di mediamorfosi (Fidler 1997).

Diventa perciò impossibile fissare la natura dei (nuovi?) media in un nuovo paradigma di analisi, che rimangono molteplici e cangianti (Bolter e Grusin 2002:12).

La linea delle ri-mediazioni *remediations*

Secondo de Keckerhove, bisogna uscire dal sistema dei media per poter vedere in prospettiva la forma vera dei media precedenti (de Keckerhove 1998 in Bolter-Grusin, 2002:12):

La forma vera della radio è stata rivelata dalla tivú. La forma della tivú è divenuta manifesta soltanto dopo l'invenzione del computer. La forma del computer è già possibile comprenderla meglio perché siamo entrati nel mondo delle Reti. La forma delle Reti, invece, non è ancora visibile, perché non c'è nessun medium piú avanzato delle Reti.

Partiamo dalle tecnologie caratterizzanti di Bolter

Riprendiamo la nozione di Jay David Bolter (1984:19) di **tecnologia caratterizzante**:

Una tecnologica caratterizzante sviluppa legami non solo metaforici con la scienza, la filosofia o la letteratura di una cultura; è sempre disponibile a fungere da metafora, ad esempio, da modello o da simbolo. Questa tecnologia assomiglia a una lente d'ingrandimento che raccoglie e focalizza idee apparentemente disparate.

Metafore e tecnologie caratterizzanti (esempi)

Riprendiamo alcuni esempi di Bolter:

- il **vaso** è la metafora alla base del demiurgo di Platone;
- gli **orologi fatti con le ruote dentate** sono la metafora alla base dell'idea di Dio come orologiaio, e Natura come orologio, proposta da molti filosofi naturali del Seicento;
- il **vapore** è la metafora alla base della concezione delle correnti irrazionaliste della filosofia continentale dell'Ottocento.

Metafore e tecnologie caratterizzanti (esempi)

Riprendiamo alcuni esempi di Bolter:

- il **vaso** è la metafora alla base del demiurgo di Platone;
- gli **orologi fatti con le ruote dentate** sono la metafora alla base dell'idea di Dio come orologiaio, e Natura come orologio, proposta da molti filosofi naturali del Seicento;
- il **vapore** è la metafora alla base della concezione delle correnti irrazionaliste della filosofia continentale dell'Ottocento.

Applichiamo il modello di Bolter alla scrittura.

L'artificialità della scrittura

Dire che la scrittura è artificiale non significa condannarla, tutto il contrario: come e piú di ogni altra creazione artificiale, essa ha valore inestimabile, poiché è essenziale allo sviluppo piú pieno dei potenziali umani interiori [. . .] Le tecnologie non sono semplici aiuti esterni, ma sono anche trasformazioni interne alla coscienza (Ong 1982:124).

Il paradigma chirografico

Dai papiri egizi ai codici medioevali, l'opera non si firma. Non esiste l'autore nel senso moderno. Lo scrivano è schiavo o monaco: l'Autore è Dio, lo scrivano copia e così tramanda. Il testo si manda a memoria, la lettura è ad alta voce.

Il Potere è nell'oralità, in chi detta la Legge, non in chi lo (tra)scrive.

Il grado di coerenza (semantica) e coesione (sintattica) del testo è basso perché poco importante. L'importanza del testo sta nella sua esecuzione, nella sua performatività.

Lo spazio chirografico

La standardizzazione dello spazio chirografico fu un processo lungo e difficile.

L'organizzazione del libro in codici (pagine piegate e legate assieme) invece che in rotoli, l'impiego della carta al posto del papiro, della tavoletta d'argilla o della pergamena, l'esistenza di un alfabeto e di una calligrafia comuni alla maggior parte dello spazio europeo, li dobbiamo a Carlo Magno e alla riforma della calligrafia imposta autoritariamente da Alcuino (Lévy, 2000; Goody in Bocchi-Ceruti, 2002).

Il paradigma tipografico

La stampa a caratteri mobili rende il testo importante di per sé, e quindi prende importanza l'autore. Senza Gutenberg, Lutero non avrebbe potuto diffondere le sue idee.

Il potere è l'autor-ità, è nell'autor-e, autor-evole e autor-itario.

Nasce la gabbia tipografica. Vengono dati i numeri alle pagine. Dato la sua prigione, il testo ricerca l'autosufficienza: nasce il metatesto, la riflessione del testo su se stesso. Coerenza e coesione diventano fondamentali.

I rimandi esterni al testo hanno i loro spazi ristretti e riservati: note, indici, bibliografia.

Michel Foucault e la funzione autoriale

La funzione-autore è legata al sistema giuridico e istituzionale che racchiude, determina, articola l'universo dei discorsi; essa non si esercita uniformemente e nella stessa maniera su tutti i discorsi, in tutte le epoche e in tutte le forme di civilizzazione; non è definita dall'attribuzione spontanea di un discorso al suo produttore, ma da una serie di operazioni specifiche e complesse (Foucault 1969).

Il libro assume il significato *tout court* di testo, e lo sacralizza.

La nascita dell'interfaccia standard tipografica

Fu Erasmo da Rotterdam il primo a numerare le pagine in analogia ai versetti della Bibbia.

Questo consolidò il percorso di lettura lineare avanti-indietro, sinistra-destra, alto-basso che è rimasto principe fino alla rivoluzione informatica.

Naturalmente, non è vero che lo spazio tipografico è puramente lineare: le pagine sono piegate e divise in unità standard (non sono rotoli) e l'invenzione dell'indice coeva a quella del numero di pagina danno la possibilità di letture non lineari.

Il testo reso riproducibile

La scrittura per sua natura si concentra sulla conservazione statica e immutabile delle informazioni. La stampa aggiunge a questo la moltiplicazione di copie identiche, il che permette a lettori separati nel tempo e nello spazio di condividere il riferimento alle stesse informazioni (Eisenstein 1980).

Il sapere umanistico come noi lo conosciamo è un “effetto culturale della stampa” (McLuhan), e ha comportato il sorgere del sapere umanistico e dell’attività critica, possibile perché la (relativa) facilità di riproduzione affranca lo studioso dalla necessità di copia.

Lo spazio tipografico

La standardizzazione dello spazio tipografico fu un cammino difficile, ma meno lungo di quello intrapreso per lo spazio chirografico.

Fu Aldo Manuzio, stampatore veneziano, a promuovere l'*in octavo*, la piegatura del foglio in otto anziché in due (*in folio*), e inventò il carattere corsivo o *italico*, che permette la presenza di una maggior quantità di caratteri sulla pagina stampata rispetto al tondo, mantenendo la leggibilità. Inoltre eliminò glosse e note tipiche dei codici medievali.

Manuzio produsse libri maneggevoli e di argomento originale, affrancando il libro stampato dal modello del codice medievale seguito da Gutenberg. Era nato il libro moderno.

Ted Nelson

Theodor Holm Nelson (classe 1937) è filosofo americano di origine norvegese. Basandosi sui lavori di Vannevar Bush, George Orwell e Douglas Engelbart, Nelson (1990) riflette sulla rivoluzione della conoscenza portata dai calcolatori, in particolare sulla nuova scrittura, che chiama **ipertesto** e **ipermedia**.

Lancia il suo progetto Xanadu negli anni 1960: la sua scrittura, visionaria e profetica, non viene compresa immediatamente. Nel 1977 è uno dei rivenditori dell'Apple I e nel 1978 viene chiamato dall'IBM come consulente per lanciare il nuovo prodotto, chiamato *personal computer*.

Attualmente è *visiting fellow* all'Università di Oxford dove lavora sul prototipo di una nuova interfaccia che superi la metafora della scrivania e della carta, chiamata ZigZag.

Il sogno dell'ipertesto globale: Xanadu

Nelson teorizza l'ipertesto come qualcosa di **aperto, globale e collaborativo**:

Altri possono fare aggiunte a qualunque produzione o riutilizzare le parti come credono opportuno. Chiunque può fare cose nuove con lo stesso materiale liberamente [...]. Il primo autore ed editore hanno il diritto di mostrarvi qualcosa a modo loro; ma gli utenti hanno il diritto di utilizzare queste cose come preferiscono [...], a patto che il contesto originario sia conservato, dev'essere consentito di sviluppare nuovi contesti (Nelson 1990:3/12).

Xanadu, “una sorta di Biblioteca di Alessandria del sapere contemporaneo” (Lévy 2000:35), è il nome del progetto:
<http://www.xanadu.net>.

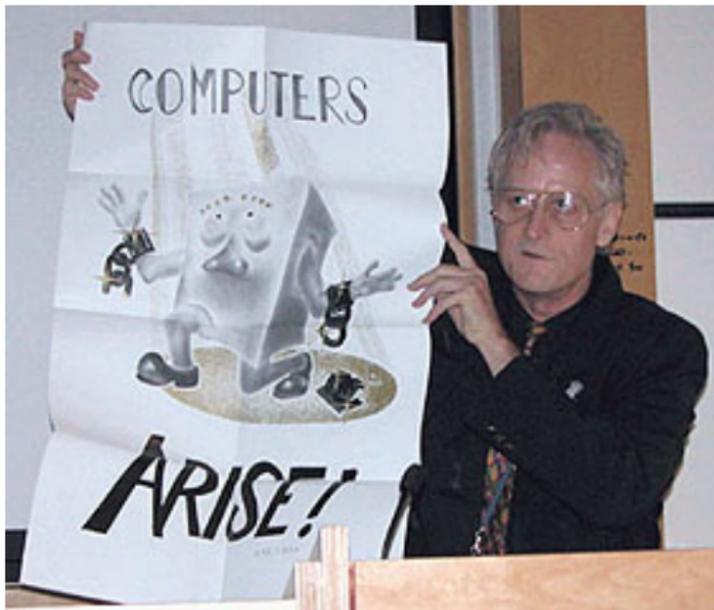


Figura: Ted Nelson alla conferenza *Hypertext* (2003)

Xanadu, ambiente iper-mediale universale aperto

Il sistema ipertestuale Xanadu è una forma di archivio: ogni documento può essere originale o un insieme di transclusioni mediante rimandi (come gli odierni link, ma tipati). L'insieme di transclusioni e rimandi tipati costituiscono le *dimensioni concettuali del documento*.

Oltre a queste, un documento è costituito dalla sua *storia* – i.e. le versioni precedenti del documento stesso – e dalle *versioni alternative* dello stesso, mediante l'evidenziazione delle differenze tra un documento e l'altro.

Bush, Vannevar PUB As We May Think	NELSON, Theodor H PUB A File Structure for the Com	LN PRV HYPERTEXT BOOK Notes
the essential feature of the memex. The process of tying two items together is the important thing.	filing arrangements. Bush stressed his file's ability to store related materials in <i>associative trails</i> , lists or chains of documents joined together.	TRAIL STRUCTURE Bush's original "trail" appears to be sequential, but with cross-trails allowed. I'm not sure whether this maps to any of the newer forms, but we ought to check this.
When the user is building a trail, he names it, inserts the name in his code book, and taps it out on his keyboard. Before him are the two items to be joined, projected onto adjacent viewing positions. At the bottom of each there are a number of blank	"When the user is building a trail, he names it, inserts the name in his code book.	THE PROBLEM OF DISSENT Dissent may be incorporated. The system imposes no assumption of completeness or
	n, Theodor H PUB It Out of Our System em, and the newcomer hear of them less. ents may be incorporated. system imposes no assumption of completeness or	infallibility on its constituent parts. inconsistencies will

I bordi dei riquadri di testo possono essere usati per indicare materiale collegato adiacente (o disponibile), materiale equivalente, o citazioni. Le cornici grafiche possono indicare altre relazioni coi loro vicini.

Figura: Il front-end di Xanadu: la Cornice Q (Nelson 1990: 4/69)

L'obiettivo di Xanadu

Il concetto alla base di Xanadu è globale e sconfinato; una visione globale di dati correlati, pubblicati sugli schermi dei computer, ed anche un nuovo modo di trattare questi dati, un modo per mettere ordine nella attuale confusione di file scorrelati nella quale ciascuno di noi sguazza [. . .] La novità nel trattare i dati sta nel rappresentare le loro vere correlazioni (Nelson 1990:1/21).

Caratteristiche dell'ipertesto secondo Nelson

- **il problema dell'inquadramento.** nel caso di raccolte documentali molto grandi, l'estrazione di sotto-raccolte implica la disattivazione di alcuni rimandi, la creazione di *contesti chiusi arbitrari*.
- **alternative complesse e confronti.** aiutare l'utente a capire la complessità, evidenziando versioni differenti, strutture parallele, progetti alternativi.
- **tipi di collegamento** quando i collegamenti diventano numerosi, diventa necessario tipare i collegamenti per evitare confusione.
- **controllo di versione** il sistema deve tenere traccia della storia del singolo documento, registrando le differenze ed evidenziando le parti corrispondenti.

Problemi di realizzabilità di Xanadu

Lévy (2000:35) individua tre problemi che hanno impedito a Xanadu di imporsi:

- ① **progettabilità di una base di dati sterminata** non è chiaro come riuscire a realizzare in pratica un sistema che registri associazioni libere.
- ② **mancanza di uno standard stabile** i protocolli di Xanadu sono in continua evoluzione, in particolare il FEBE, per l'archiviazione documentale.
- ③ **mancanza di competenze sui contenuti** chi possiede le competenze necessarie per organizzare, ritagliare, mettere in scena, accompagnare documenti di contenuto arbitrario?

Centralità dell'ipertesto come modello cognitivo

L'ipertesto *esteriorizza* le reti di significato transitorie Lévy (2000:29–32):

[...] una metafora che vale per tutte le sfere della realtà in cui sono in gioco dei significati [...] Gli attori della comunicazione producono [...] l'universo di senso che li unisce o li oppone. Ma la stessa operazione di costruzione del contesto si rigioca su scala di una micro-politica interna ai messaggi [...] i giocatori non sono più persone ma elementi di rappresentazione [...] l'interazione delle parole crea delle reti di significato transitorie nella mente di un ricevente.

I sei principi dell'ipertesto (segue)

1. *Principio di metamorfosi* La rete ipertestuale è continuamente in costruzione e rinegoziazione.
2. *Principio di eterogeneità* I nodi e i legami di una rete ipertestuale sono multimediali, multi-modali, come nella mente umana.
3. *Principio di molteplicità e di incastro delle scale* Qualsiasi nodo o qualsiasi legame può rivelarsi all' analisi composto di una rete, e così di seguito, indefinitamente.

I sei principi dell'ipertesto (fine)

4. *Principio di esteriorità* La rete dipende da un esterno indeterminato: aggiunta di nuovi elementi, collegamento con altre reti, eccitazione di elementi terminali (ricevitori).
5. *Principio di topologia* tutto funziona secondo la prossimità, la vicinanza. La rete non sta nello spazio, è lo spazio.
6. *Principio di mobilità dei centri* La rete ha diversi centri continuamente in movimento, che corrono a disegnare, come punti luminosi, cangianti paesaggi del senso.

Nascita dello Xerox PARC (Allan 2001)

Nel **1970** la Xerox decide di entrare nel mercato dell'informatica nel settore delle macchine per l'ufficio, e contestualmente viene fondato il **PARC** (*Palo Alto Research Center*), in California, che assorbe la Scientific Data System (SDS, di cui non abbiamo parlato).

Al PARC il progetto di ricerca principale è sviluppare un **computer ad uso personale**, sulla base delle idee di Douglas Engelbart, Ivan Sutherland e Alan Kay.

Alan Kay

Alan Curtis Kay (classe 1940) nel 1969 conclude la sua tesi di dottorato dal titolo *The Reactive Engine*, in cui descrive un linguaggio di programmazione chiamato FLEX. Nei due anni successivi estende il concetto all'hardware, proponendo una macchina chiamata *Dynabook*.

Entrato al PARC nel 1972, entra in contatto con le idee di Engelbart e diventa figura chiave del design di un computer personale completamente innovativo, chiamato **ALTO**, la cui prima unità viene completata nel 1973.



Figura: Xerox Alto

Il progetto ALTO

Il linguaggio di programmazione di ALTO era inizialmente il MESA, molto simile al Pascal di Niklaus Wirth, mentre il sistema operativo era scritto in un linguaggio simile al C (vediamo oltre). ALTO si collegava ad ARPANET mediante una scheda **Ethernet**, e un protocollo a pacchetti chiamato PUP.

Butler Lampson e Charles Simonyi scrivono un word processor chiamato **Bravo**, dove viene introdotta l'idea del **WYSIWYG** (*What You See Is What You Get*). Alan Kay inventa un linguaggio di programmazione nuovo, chiamato **Smalltalk**.



Figura: Alan Kay in una foto d'epoca

How to classify programming languages (Kay 1993)

Programming languages can be categorized in a number of ways: imperative, applicative, logic-based, problem-oriented, etc. But they all seem to be either an “agglutination of features” or a “crystallization of style.” COBOL, PL/1, Ada, etc., belong to the first kind; LISP, APL– and Smalltalk–are the second kind. It is probably not an accident that the agglutinative languages all seem to have been instigated by committees, and the crystallization languages by a single person.

Why Smalltalk (Kay 1993)

Smalltalk's design—and existence—is due to the insight that everything we can describe can be represented by the recursive composition of a single kind of behavioral building block that hides its combination of state and process inside itself and can be dealt with only through the exchange of messages. Philosophically, Smalltalk's objects have much in common with the monads of Leibniz and the notions of 20th century physics and biology.

What is Smalltalk (Kay 1993)

In computer terms, Smalltalk is a recursion on the notion of computer itself. Instead of dividing “computer stuff” into things each less strong than the whole—like data structures, procedures, and functions which are the usual paraphernalia of programming languages—each Smalltalk object is a recursion on the entire possibilities of the computer. Thus its semantics are a bit like having thousands and thousands of computer all hooked together by a very fast network.

Orientazione a oggetti (OO) e utenti

Smalltalk è il primo linguaggio di programmazione che porta l'idea dell'OO a piena maturazione, e Alan Kay lo intende per l'**utente**, vale a dire **programmatori non esperti**, come per esempio i bambini.

Alan Kay si riferisce esplicitamente a pedagoghi costruttivisti quali Maria Montessori o Seymour Papert, ed è fautore di una **democratizzazione della programmazione software**, che avrà luogo con il personal computer.

Donald Knuth (da: Bossi–Di Muro 2006)

Donald Ervin Knuth (classe 1938) ottiene il PhD in matematica nel 1963. Nel 1968 entra a Stanford, poi ottiene il Turing Award e altre onorificenze. La sua opera principale è **The Art of Computer Programming**, un compendio di analisi degli algoritmi da Euclide in avanti (primo volume: 1968; secondo: 1969; terzo: 1973).

Nel preparare la seconda edizione, si accorge che non esiste uno strumento di tipografia digitale adeguato, così se lo inventa: nasce **T_EX** (leggi: **tek, sta per il greco tekné**), per la composizione, e Metafont, per lo sviluppo dei caratteri.



Figura: Donald Knuth nel 2005

Donald Knuth (di qui: Bossi–Di Muro 2006)

Donald Ervin Knuth (classe 1938) ottiene il PhD in matematica nel 1963. Nel 1968 entra a Stanford, poi ottiene il Turing Award e altre onorificenze. La sua opera principale è **The Art of Computer Programming**, un compendio di analisi degli algoritmi da Euclide in avanti (primo volume: 1968; secondo: 1969; terzo: 1973).

Nel preparare la seconda edizione, si accorge che non esiste uno strumento di tipografia digitale adeguato, così se lo inventa: nasce **T_EX** (leggi: **tek, sta per il greco tekné**), per la composizione, e Metafont, per lo sviluppo dei caratteri.

La situazione della tipografia digitale nel 1976

All'epoca, i programmi orientati alla produzione letteraria erano:

- proprietari;
- molto costosi;
- spesso limitati a specifici hardware;
- non precisi, nel senso che la stessa espressione stampata in due punti diversi non sempre appariva uguale, soprattutto se il programma era di tipo WYSIWYG;
- non standard, nel senso che lo stesso documento processato su due sistemi diversi non appariva allo stesso modo.

Vantaggi di T_EX

- T_EX compone facilmente formule matematiche innestate nel testo sotto forma di caratteri ASCII, anche esportate da **MATHEMATICA** o **MAPLE**
- T_EX è portabile: gira su qualsiasi computer di quarta generazione (dal 1977 in avanti).
- T_EX è software libero *ante litteram*.

T_EX è stabile e definitivo dal 1982, con alcune aggiunte nel 1989: la versione definitiva è la 3.141592.

Da T_EX a L^AT_EX

Composto di sole 300 primitive e 600 istruzioni derivate, T_EX si è rivelato ostico da usare per utenti non avanzati.

Nel **1985** Leslie Lamport (ricercatore statunitense sulle logiche temporali prima per Digital, oggi per Microsoft) scrive delle macro per T_EX che chiama L^AT_EX dal suo cognome, che oggi è lo standard dell'editoria scientifica ma non solo mondiale.



Figura: Leslie Lamport alla IFIP, Udine 2006 (fonte: andrej.com)

L'informatica di sinistra, una breve stagione. . .

Nel 1975 il Pci ottiene un successo elettorale eccezionale alle amministrative. Mario Bolognani diventa “rivoluzionario di professione”, entrando nella sezione di Varese e diventa l'uomo di riferimento del partito per l'informatica. Nel 1979, in un'Italia scioccata dal rapimento Moro, Bolognani elabora un ambizioso piano di informatizzazione della Pubblica Amministrazione (Bolognani 2004:68–70), i cui temi sono:

- usare architetture distribuite in rete, sia hardware che software;
- applicare l'ingegneria del software;
- gestire i contenuti informativi;
- scrivere a tal fine applicativi appropriati.

Ma Giorgio Napolitano sostiene Giovanbattista Gerace, professore di Pisa collaboratore alla CEP, che invece intende costituire un polo nazionale dell'informatica attorno alla Finsiel.

L'informatica di sinistra, una breve stagione. . .

Nel 1975 il Pci ottiene un successo elettorale eccezionale alle amministrative. Mario Bolognani diventa “rivoluzionario di professione”, entrando nella sezione di Varese e diventa l'uomo di riferimento del partito per l'informatica. Nel 1979, in un'Italia scioccata dal rapimento Moro, Bolognani elabora un ambizioso piano di informatizzazione della Pubblica Amministrazione (Bolognani 2004:68–70), i cui temi sono:

- usare architetture distribuite in rete, sia hardware che software;
- applicare l'ingegneria del software;
- gestire i contenuti informativi;
- scrivere a tal fine applicativi appropriati.

Ma Giorgio Napolitano sostiene Giovanbattista Gerace, professore di Pisa collaboratore alla CEP, che invece intende costituire un polo nazionale dell'informatica attorno alla Finsiel. Ma non se ne fa nulla.

E il mondo cattolico?

Mentre la DC non ritiene l'informatica importante, il Vaticano sí: il teologo Roberto Busa (classe 1913), gesuita, è il pioniere della linguistica computazionale in Italia. Nel 1946 scrive la sua tesi di laurea, dove illustra l'idea di una verifica puntuale e integrale del lessico di Tommaso d'Aquino mediante computer. Nel 1949 è a New York e convince Thomas Watson Sr., l'amministratore delegato della IBM, a finanziare l'impresa.

Per 30 anni Busa lavora con gli ingegneri IBM: nel **1980** viene data alle stampe l'**Index Thomisticus**, opera a stampa in 56 volumi, dove l'opera omnia di Tommaso viene lemmatizzata in forma ipertestuale.

E il mondo cattolico?

Mentre la DC non ritiene l'informatica importante, il Vaticano sí: il teologo Roberto Busa (classe 1913), gesuita, è il pioniere della linguistica computazionale in Italia. Nel 1946 scrive la sua tesi di laurea, dove illustra l'idea di una verifica puntuale e integrale del lessico di Tommaso d'Aquino mediante computer. Nel 1949 è a New York e convince Thomas Watson Sr., l'amministratore delegato della IBM, a finanziare l'impresa.

Per 30 anni Busa lavora con gli ingegneri IBM: nel **1980** viene data alle stampe l'**Index Thomisticus**, opera a stampa in 56 volumi, dove l'opera omnia di Tommaso viene lemmatizzata in forma ipertestuale.

<http://www.corpusthomisticum.org/>



Figura: Padre Roberto Busa con alle spalle la sua opera (2006)

Grazie. Domande?



Potete scaricare questa presentazione qui:

<http://www.slideshare.net/goberiko/>

© (CC) (BY) (NC) (SA) Federico Gobbo 2009 di tutti i testi. Pubblicato in Italia.
Attribuzione – Non commerciale – Condividi allo stesso modo 2.5

© delle figure degli aventi diritto. In caso di violazione, scrivere a: federico.gobbo@uninsubria.it.