

quaderni di comunicazione audiovisiva

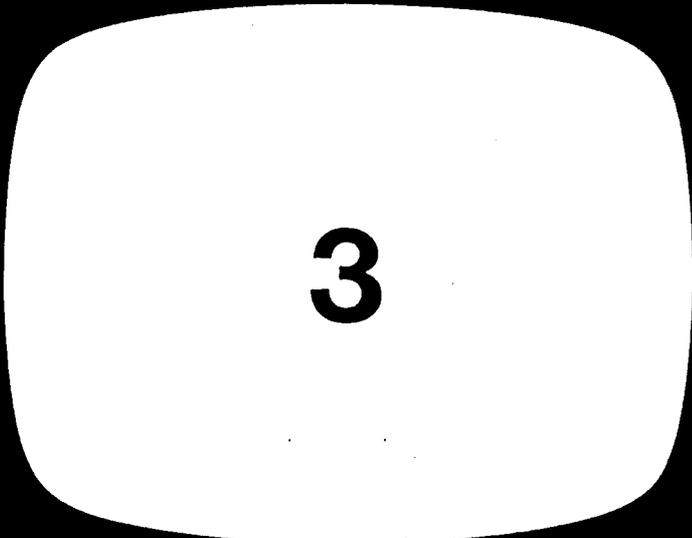
Rivista trimestrale edita in collaborazione con:

Laboratorio Audiovisivo Universitario
Università di Ginevra,

Cattedra di Metodologia e Didattica degli
Audiovisivi - Università di Padova,

Scuola di Comunicazioni Audiovisive
Enaip Regione Lombardia - Pavia.

Anno 1 - n. 3 1984



3

editrice mcm

editoriale

LUCIANO GALLIANI

Tecnologie educative o tecnologie della comunicazione
educativa?

pag.

studi

GIOVANNI LARICCIA	Progettazione educativa e complessità informatica	pag.	12
PETER KNOPF	Educazione e scuola nella società telematica	pag.	22
ANGELO D'ALESSANDRO	Documento e fiction nei programmi culturali audiovisivi	pag.	36

proposte

MARCELLO GIACOMANTONIO	Gaming simulation e comunicazione educativa	pag.	56
FIORINO TESSARO	Architettura di un sistema informatizzato del software didattico	pag.	74

esperienze

SILVIA CECCHI	I brani musicali nelle interazioni con le sequenze di immagini	pag.	86
CATHERINE BERDONNEAU GERARD BOSSUET	Ambiente "LOGO" e comportamento dei bambini	pag.	100

rubriche

	1ª Rassegna Europea "Audiovisivi e Scuola"	pag.	108
	1° Festival Internazionale Audiovisivi d'Europa	pag.	113
	Un "progetto mass-media" per l'associazionismo	pag.	116
	recensioni	pag.	121

sintesi

	summaries - zusammenfassungen	pag.	126
--	-------------------------------	------	-----

Architettura di un sistema informatizzato del software didattico

1. PREMESSA

Le analogie e le possibili integrazioni riscontrabili tra informatica e didattica, permettono di ipotizzare una 'terza via' per l'accesso alla cultura informatica al di là dell'Istruzione Programmata assistita dall'elaboratore, ma anche dell'eccessiva specializzazione richiesta dai linguaggi di programmazione. Questa 'terza via', che implica l'utilizzo di tecnologie informatiche nell'ottica dei modelli teorici e delle metodologie progettuali comuni, deve svilupparsi in un progetto globale per la gestione informatizzata e dinamica delle risorse hardware e software ad uso didattico. Considereremo qui la progettazione hardware semplicemente in funzione di quella software, determinate ai fini del nostro lavoro.¹

Prima di illustrare le linee operative del progetto, è necessario specificare i due elementi-chiave del metodo d'insegnamento, presenti attualmente per lo più nella sola fase di programmazione didattica: la struttura logica di un corso e l'algoritmo didattico.

La *struttura logica di un corso* (fig. A/0) riguarda l'ordine in cui devono essere esposti gli argomenti essenziali, ordine che determina i rapporti reciproci tra le sezioni del corso stesso e quelli con i corsi 'affini'.

Il corso viene modularmente diviso in sezioni, le sezioni in argomenti, questi in capitoli, che vanno a loro volta divisi in unità didattiche (UD).

In linea generale, la struttura logica di un corso deve determinare: l'estensione del contenuto, l'ordine pedagogicamente razionale dell'insegnamento, il rapporto logico reciproco tra le sezioni, il legame logico tra questo corso ed altri. In breve, deve rispondere alla domanda: *'cosa' si deve insegnare?*

L'algoritmo didattico stabilisce lo scopo della soluzione del compito, l'informazione iniziale necessaria per risolverlo, il sistema di regole per la rielaborazione dell'informazione iniziale e l'ordine di applicazione di queste regole. Deve perciò rispondere essenzialmente alla domanda *'come' si deve insegnare?*

Come si vede la struttura logica e l'algoritmo didattico di un corso hanno significati diversi ma che si integrano a vicenda. Si deve quindi cominciare con il determinare la struttura logica di un corso elaborando successivamente l'algoritmo didattico rispetto al quale la struttura appare come informazione iniziale.

La fig. A/1 illustra lo schema della struttura fondamentale dell'algoritmo didattico in una rielaborazione del modello di J. P. Gornjunov².

Queste le fasi in cui si sviluppa l'algoritmo.

Determinata la struttura logica del corso, il docente presenta (quadro 1) il contenuto o parte del contenuto «UD (i)» relativa a una prima unità didattica: in questa fase è determinante l'esposizione completa ed essenziale del sistema di regole e/o punti-chiave in cui si sviluppa l'UD(i). Segue l'analisi del grado di apprendimento di UD(i) (quadro 2) con una serie di domande di controllo proposte all'allievo, elaborate per stabilire anche il grado di apprendimento delle precedenti UD(i-k) più strettamente correlate a UD(i). Se UD(i) è stata assimilata si passa all'esposizione della successiva UD(i+1) (quadro 3), altrimenti ai quadri 4 e 5, nei quali si chiarisce rispettivamente ciò che non è stato appreso in UD(i) e nelle precedenti UD(i-k). Con le domande di controllo del quadro 6 l'allievo è messo in condizione di ricordare le UD(i-k) già studiate, ma

L'informatica sta coinvolgendo tutti i settori della formazione: è un momento estremamente delicato in quanto la "massificazione" culturale, metodologica e tecnologica dell'informatica richiede la massima chiarezza ed obiettività circa le enormi possibilità offerte, ma anche i limiti ad essa connaturati.

Non si tratta di informatizzare la scuola, ma di offrire al docente una struttura, nel contempo agile e potente, entro cui sviluppare e adattare il proprio intervento didattico in funzione delle peculiarità del gruppo di apprendimento.

Il modello illustrato nel presente articolo vuole essere una prima risposta operativa alla possibilità di progettare un intervento formativo secondo paradigmi informatici nel rispetto della professionalità del docente e della creatività dell'allievo.

in parte dimenticate. Sulla base delle risposte, il quadro 7 analizza il grado di apprendimento di UD(i-k). Se questa non è stata assimilata si ripetono gli argomenti non appresi (quadro 8) e, conseguentemente, si torna al quadro 6. In caso contrario si ricontrolla il grado di apprendimento di UD(i) (quadro 2). Se l'analisi risulta affermativa si procede a esporre la successiva UD. Se invece l'UD(i), a differenza delle precedenti UD(i-k), non è stata appresa, bisogna ripetere l'esposizione di UD(i) nel quadro 1. I tre misuratori ('counter') — nn(i), n(i), n(i-k) — volutamente inseriti nello schema, necessari per individuare i punti in cui si riscontra da parte dell'alunno/i una maggior difficoltà di apprendimento, possono seguire l'iter di apprendimento del singolo alunno e sviluppare un profilo individuale utile ai fini della valutazione. I counter possono anche essere considerati collettivamente, permettendo una 'standardizzazione', sia della struttura logica del corso sia della struttura interna dell'unità didattica.

Sulla base della rappresentazione grafica della struttura logica di un corso (Fig. A/0) e dello schema inerente la struttura fondamentale dell'algoritmo didattico (Fig. A/1), possiamo sviluppare l'architettura del nostro progetto alla stregua di un sistema informativo interattivo e dinamico. (Fig. A/2).

Questi i macro-archivi ('file') impegnati:

- archivi del software esistente o disponibile
- archivi domande-risposte
- archivi valutazioni individuali e collettive.

L'archivio-software, strutturato come un Data Base (e cioè con tutte le caratteristiche immesse utilizzabili come codici-chiave per la ricerca), è formato da due file: l'M-SOFT e il DIDACT.

2 - L'ARCHIVIO DEL SOFTWARE

L'archivio M-SOFT memorizza i dati essenziali ('campi') inerenti ciascun "oggetto software" ad uso didattico ('record'). In Figg. A/3a e A/3b vengono descritti i possibili campi, la loro tipologia e il loro assemblamento in un record M-SOFT.

I dati relativi all'autore (1), al titolo (2), all'eventuale collocazione di articoli in riviste o testi (3) e i dati relativi all'edizione e/o produzione — denominazione (4), luogo (5) e anno (6) — rispecchiano una normale procedura bibliografica e pertanto non hanno bisogno di particolari spiegazioni.

Il codice-software (7) indica il tipo di materiale archiviato. Ad esempio: AF0C1 può indicare software di tipo Audiovisivo-Fotografico-Colori, mentre BM0S4 denoterà software di tipo Bibliografico-Monografia-Storica, ecc. (8) A riguardo, va precisato che le codificazioni vanno preventivamente costruite e uniformate da tutti gli utenti del sistema. È opportuno, inoltre, utilizzare sigle alfanumeriche per la facilità di memorizzazione da parte dell'operatore.)

Il codice successivo dovrà analogamente specificare il tipo-hardware, e cioè il supporto utilizzabile, e la sua reperibilità. Determinati hardware non possono essere trasportati, fanno parte di dotazioni di particolari laboratori e pertanto sono vincolati dalla capacità di farli funzionare o semplicemente dall'orario della loro utilizzazione.

I codici di catalogazione (9) e di collocazione (10) seguono le normali metodologie di inventariazione già presenti nelle scuole.

Alla fine del record abbiamo riservato due aree libere per eventuali codici utilizzabili dal docente per ulteriori specificazioni del software. Va comunque precisato che tali codici devono riguardare l'intero software e non solo parte di esso.

Come precedentemente accennato, nel file M-SOFT si memorizzano i dati inerenti tutti i tipi di software, il quale può essere:

— *stampato*: testi, libri, riviste, monografie, enciclopedie, ecc.

— *audiovisivo*: fotografie, diapositive, dischi e registrazioni audio-visive in genere

— *informatico*: routines didattiche, corsi programmati, ecc.

Va sottolineato al riguardo che tutto il materiale audiovisivo ed infomatico deve essere integrato dal relativo manuale operativo al fine di uniformare la ricerca all'interno dell'archivio.

Fig. A/0 Rappresentazione grafica della struttura logica di un corso

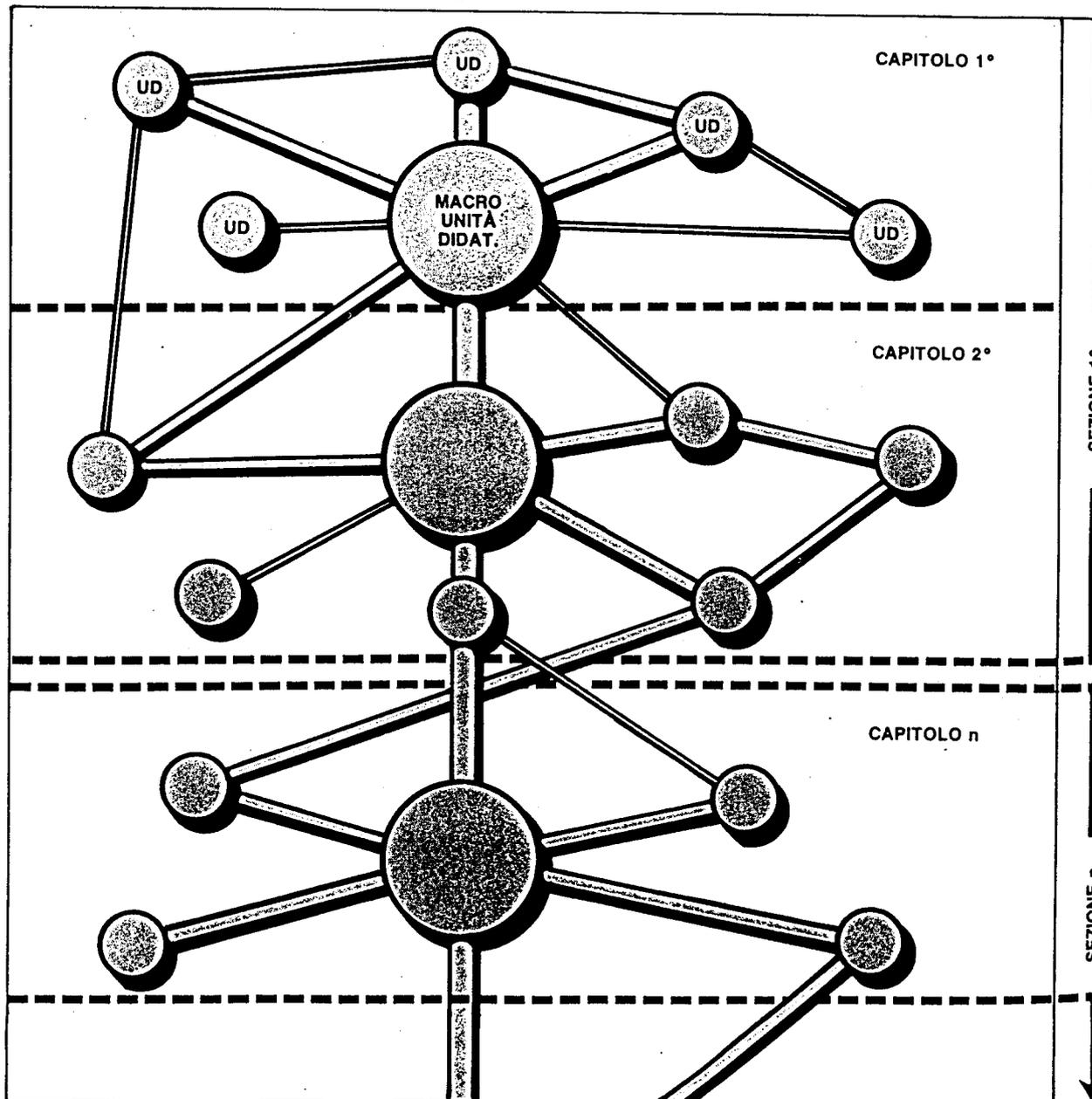
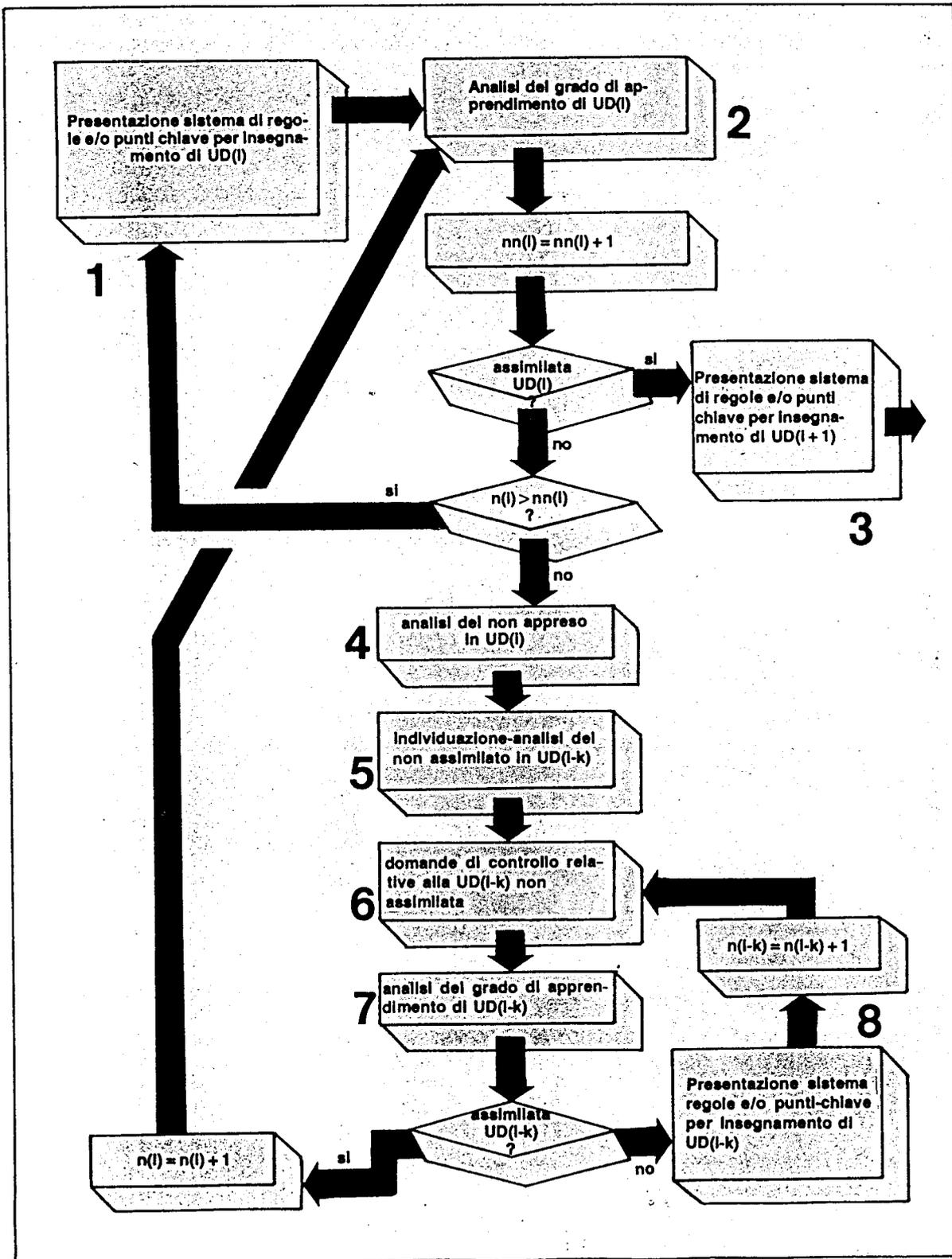


Fig. A/1 Struttura fondamentale dell'algoritmo didattico. (Rielaborazione da J. P. Gorjunov)



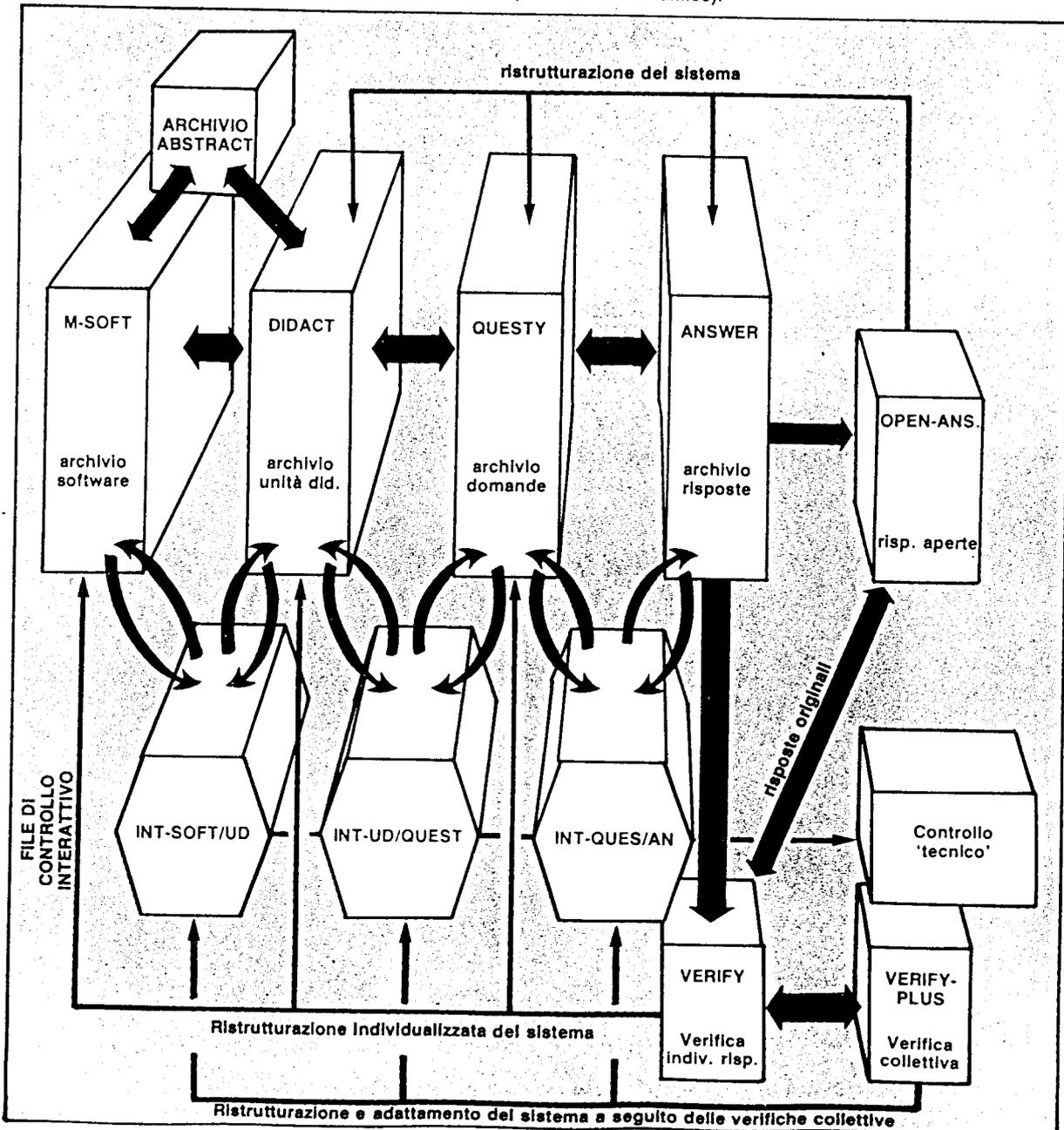
3 - L'ARCHIVIO DELLE UNITÀ DIDATTICHE

Ogni record M-SOFT può 'esplodere' in uno o più record presenti nell'archivio DIDACT. Il singolo software, infatti, può interessare una o più unità didattiche e nell'ambito di ciascuna con diversi gradi di

coinvolgimento. È facilmente intuibile come, a differenza dell'M-SOFT con inserimenti di natura quasi 'amministrativa', la creazione di quest'ultimo file richiede un maggior impegno da parte del docente.

Nelle Figg. A/4a e A/4b vengono illustrati i possibili campi, la loro tipologia ed il loro assemblamento nel record DIDACT.

Fig. A/2 Architettura di un sistema Informativo-didattico (Interattivo e dinamico).



Il primo dato deve permettere l'“aggancio” al relativo record M-SOFT. Il secondo campo specifica globalmente l'unità didattica interessata. Ad esempio: E4ARF1 può corrispondere ad una unità didattica predisposta per una classe quarta elementare-aritmetica-frazioni-primo approccio, mentre S5STF6 può corrispondere allo studio sugli aspetti economici del fascismo per una classe quinta superiore.

Segue la localizzazione della parte del software interessante l'unità didattica. Questo campo può essere descritto in pagine (da-a) o in sezioni, capitoli, paragrafi, ecc.

Il quarto dato denota il livello di coinvolgimento software-unità didattica. Il software, infatti, può risultare centrale e determinante ai fini dello sviluppo dell'unità didattica, ma può anche, a vari livelli, risultare propedeutico, riassuntivo, marginale, di legame logico con UD precedenti, successive o parallele, ecc.

Formulare una valutazione codificabile su un intero software può risultare arduo e spesso contraddittorio. Per questo motivo nel quinto campo va memorizzata (e quantificata) la valutazione, da parte del docente, relativa alla sola parte di software implicata nell'unità didattica e soprattutto in relazione al suo livello di coinvolgimento.

Il sesto dato specifica il codice per l'accesso all'eventuale abstract contenente i punti-chiave dell'unità didattica.

I quattro campi successivi richiedono, secondo priorità di correlazione, la descrizione delle eventuali unità didattiche collegate con quella in fase di descrizione, nello sviluppo dello specifico software.

Molti dei dati memorizzati del file DIDACT potranno a prima vista sembrare arbitrari ed è fuori d'ogni dubbio che parte di essi si riveleranno ben presto errati. In fase di codificazione, infatti, possiamo andare incontro a due tipi di errore: o per mancata ‘aderenza’ al software o per imprecise valutazioni delle capacità di apprendimento degli alunni.

Il primo tipo di errore, che potremmo definire ‘tecnico’, può essere, nella maggior parte dei casi rintracciato e corretto in automazione. Facendo interagire i campi dei due record (ad es. tutti i DIDACT inerenti un

determinato M-SOFT o, globalmente, tutte le unità didattiche — in codice — tra di loro) possiamo ottenere in output dei grafi con in evidenza:

a) le affinità per software e per unità didattiche (grappoli di UD in sequenza e con elevate correlazioni con il software) che facilitano la stesura della struttura logica del corso (Fig. 8/0);

b) le mancate ‘aderenze’ che possono presentare, invece, una doppia valenza:
— di sviluppo originale rispetto alla classica sequenza di unità didattiche o
— di vero e proprio errore di codificazione che ovviamente richiede un'immediata correzione.

Le imprecise valutazioni riguardo alle capacità di apprendimento degli alunni, a differenza degli errori ‘tecnici’, possono essere corrette solo nella fase di controllo dell'apprendimento. In realtà, in questo caso si tratta di correggere la “direzione di rotta” circa determinate codificazioni, come il livello di coinvolgimento, il giudizio, la formulazione dell'UD e la sua correlazione con altre unità didattiche. Tale operazione rientra in quella che, più avanti, definiremo come ‘modello dinamico’. La ‘correzione di rotta’, infatti, può rendersi necessaria più volte a causa dell'eterogeneità dei gruppi scolastici nel differenziarsi delle dimensioni culturali, spazio-temporali, socio-economiche, ecc.

4 - GLI ARCHIVI DOMANDE-RISPOSTE

L'archivio delle domande (QUESTY) si presenta come un grande contenitore continuamente integrato con nuovi inserimenti da parte degli operatori-docenti.

È importante che ciascun record QUESTY, accanto alla formulazione completa della domanda, contenga anche:

a) il codice-chiave primario corrispondente all'unità didattica, nel record DIDACT;

b) più codici secondari, sempre relativi al record DIDACT, cui poter far riferimento nell'attività di “transinformazione” o di transfert e nell'analisi del grado di apprendimento (quadri 2-4-6-7 in Fig. A/1);

DESCRIZIONE CAMPI FILE M-SOFT		L = 150	
1 AUTORE/I	L = 25 D = 1	ALFANUMERICO	VARIABILE
2 TITOLO	L = 75 D = 26	ALFANUMERICO	VARIABILE
3 C. RIV./SOFT	L = 4 D = 101	NUMERICO	VARIABILE
4 EDIT/PROD.	L = 12 D = 105	ALFANUMERICO	VARIABILE
5 LUOGO	L = 10 D = 117	ALFANUMERICO	VARIABILE
6 ANNO	L = 3 D = 127	NUMERICO	FISSO DI 3 CHR
7 C. SOFTWARE	L = 5 D = 130	ALFANUMERICO	FISSO DI 5 CHR
8 C. HARDWARE	L = 3 D = 135	ALFANUMERICO	FISSO DI 3 CHR
9 C. CATALOG.	L = 3 D = 138	ALFANUMERICO	FISSO DI 3 CHR
10 COLLOCAZ.	L = 3 D = 141	ALFANUMERICO	FISSO DI 3 CHR
11 COD. EVEN. 1°	L = 3 D = 144	NUMERICO	VARIABILE
12 COD. EVEN. 2°	L = 3 D = 147	NUMERICO	VARIABILE

TESS/83/PD

Fig. A/3a

TRACCIATI SCHEDA

DATA:

FILE: M-SOFT

PROGRAMMA SIST.INF.DIDATTICO

PROGRAMMATORE:

1	AUTORE/I		TITOLO								
2	TITOLO	Cod. Riva-Soft	Edizione/Produzione			Codice Software	Cod. Hard	Cod. Cat	Cod. Col	Cod. Even 1	Cod. Even 2
			Denominazione	Luogo	Anno						

Fig. A/3b

c) accanto a ciascuno dei codici appena esposti, il grado di difficoltà della domanda rapportato all'unità didattica ed al software implicati;

d) il codice relativo al record-risposta (ANSWER). Questo dato viene memorizzato in automazione, considerato che ad ogni domanda deve corrispondere almeno un record ANSWER.

Quest'ultimo, accanto all'esposizione della risposta, deve memorizzare:

- a) il codice tipo-risposta. La risposta, infatti, può essere classificata:
 - chiusa binaria (si/no)
 - chiusa ramificata univoca (una sola esatta)

- chiusa ramificata complessa (diversi livelli di esattezza)

- aperta binaria (si/no, perché)
- aperta ramificata univoca (solo una, perché)

- aperta ramificata complessa (la prima, perché...)

— (Possono essere inserite altre tipologie di risposte, ad esempio in base alla completezza, alla logicità, ecc. È comunque importante che tutti i tipi di risposta possano essere codificati.)

b) codice-chiave per l'accesso all'archivio delle risposte aperte (OPEN-ANSWER), dove vengono memorizzate anche le risposte originali degli allievi.



5 - GLI E DI

Le ris: sere me IND) al

— il fecc apprend municaz tutti i mo comunic — l'anal ducibile individu: — la pro lutazion ficamen

Il recc

— il cod ve di ac dall'alur

DESCRIZIONE CAMPI FILE DIDACT		L = 053	
1. COD. M-SOFT	L = 4 D = 1	ALFANUMERICO	VARIABILE
2. UNITA DIDAT	L = 6 D = 5	ALFANUMERICO	FISSO DI 6 CHR
3. LOC. IN SOFT	L = 10 D = 11	ALFANUMERICO	VARIABILE
4. LIV. COINVOL	L = 2 D = 21	NUMERICO	VARIABILE
5. VALUTAZIONE	L = 2 D = 23	NUMERICO	VARIABILE
6. C. ABSTRACT	L = 4 D = 25	NUMERICO	VARIABILE
7. U. D. CORR. 1°	L = 6 D = 29	ALFANUMERICO	FISSO DI 6 CHR
8. U. D. CORR. 2°	L = 6 D = 35	ALFANUMERICO	FISSO DI 6 CHR
9. U. D. CORR. 3°	L = 6 D = 41	ALFANUMERICO	FISSO DI 6 CHR
10. U. D. CORR. 4°	L = 6 D = 47	ALFANUMERICO	FISSO DI 6 CHR
TESS/83/PD			

Fig. A/4a

TRACCIATI SCHEDA

DATA:

FILE: DIDACT

PROGRAMMA SIST. INF. DIDATTICO

PROGRAMMATORE:

1	Codice M-Soft	Unità Didattica	Localizzazione in M-Soft	Coinvolgimento	Valutazione	Codice Abstr.	Unità Didattica Corr. 1	Unità Didattica Corr. 2	Unità Didattica Corr. 3	Unità Didattica Corr. 4	
0+											

Fig. A/4b

5 - GLI ARCHIVI DI VERIFICA E DI VALUTAZIONE

Le risposte di ciascun allievo devono essere memorizzate ed archiviate (VERIFY-IND) al fine di permettere:

- il feed-back nell'attività di insegnamento-apprendimento, elemento chiave nella comunicazione a scopo istruttivo presente in tutti i modelli teorici concernenti i processi comunicativi;
- l'analisi del grado di apprendimento traducibile operativamente in una valutazione individuale formativa;
- la progressione nell'apprendimento o valutazione sommativa illustrabile anche graficamente sotto forma di profilo personale.

Il record VERIFY-IND deve contenere:

- il codice identificativo dell'allunno (chiave di accesso personale conosciuta solo dall'allunno e dal docente)

- il codice della domanda (QUESTY)
- la risposta data dall'allievo e
- la valutazione (codificabile).

Il sistema elabora le informazioni relative al singolo allievo confrontandole con le risposte standard (ANSWER) o con le eventuali OPEN-ANSWER. La valutazione di tutte le risposte con l'esclusione di quelle rigidamente 'chiuse', richiedono l'intervento o, per lo meno, la supervisione interpretativa del docente. Solo quest'ultimo, infatti, può verificare il grado di esattezza, di completezza, di originalità, di impostazione procedurale ed anche di semplici 'sfumature' sempre presenti in risposte aperte.

Le valutazioni individuali vengono ripartite, in base al codice-domanda, in un'area di lavoro (Work-area). In questa fase il processo elaborativo procede completamente in automazione correlando i dati presenti nella Work-area con quelli, se esistenti, del-

le valutazioni standardizzate dell'archivio VERIFY-PLUS.

In tal caso se i coefficienti di correlazione risultano molto bassi, il problema va individuato nella presentazione e/o strutturazione dell'unità didattica (quadri 1 e 8 della Fig. A/1).

Qualora, invece, (e questo sarà il caso più frequente) non esistessero valutazioni standardizzate inerenti la specifica domanda-risposta e le valutazioni collettive manifestassero generali situazioni problematiche nell'analisi dell'apprendimento, il sistema deve permettere l'individuazione dei punti critici e la loro correzione. Questa fase qualifica il sistema come "modello interattivo e dinamico".

6 - STRUTTURE INTERATTIVE E ANALISI DINAMICA DEL SISTEMA

I contenuti e le metodologie didattiche hanno registrato, specie negli ultimi anni, continui e profondi mutamenti. Non è, pertanto, possibile pensare al sistema formativo come ad un sistema statico ed immutabile.

D'altra parte anche da un 'fotogramma' della situazione scolastica, istituzionalizzata e non, emergono profonde divergenze nel modo di intendere e di praticare "il far scuola". Non possiamo, in questa sede, analizzare il problema dal punto di vista della variabilità all'interno della professionalità docente. Ci limitiamo, invece, a motivare tali divergenze, da un lato, con l'eterogeneità della provenienza culturale, socio-economica, geografica, ecc. dei gruppi scolastici e, dall'altro, con le differenze individuali nelle capacità di apprendimento.

Il nostro progetto non può non tenere in debito conto il 'dinamismo' e la variabilità caratterizzanti il sistema formativo. Per questo motivo abbiamo inserito delle strutture di interazione (o file di lavoro) tra i macro-archivi che, costruiti in automazione, possono essere rapidamente riconvertiti ogni qualvolta si vengono a modificare i dati di raccordo tra gli archivi. I tre file,

— INT-SOFT/UD (di interazione tra il software e le unità didattiche)

— INT-UD/QUES (di interazione tra le UD e le domande)

— INT-QUES/AN (di interazione tra le domande e le risposte),

dovranno, in ultima analisi, permettere output grafici (su video o su plotter) e/o tabulati (matrici bi-o tri-dimensionali) di facile lettura per l'individuazione delle eventuali situazioni problematiche che si vengono a creare a seguito di frequenti risposte errate dal punto di vista collettivo (VERIFY-PLUS).

In pratica, se una risposta, o un grappolo di risposte, possiede un elevato grado di errore è possibile rintracciare il punto critico risalendo gerarchicamente i file di lavoro:

— l'interazione QUESTY-ANSWER (la domanda presenta una formulazione contraddittoria o i codici-chiave, primari e secondari, non corrispondono alle UD memorizzate o, infine, risulta errato il grado di difficoltà);

— l'interazione DIDACT-QUESTY. È questa l'interazione fondamentale del sistema. Infatti, accanto ai possibili errori appena descritti, deve valutare i dati inerenti le unità didattiche implicate, anche con l'analisi dei relativi abstract;

— l'interazione M-SOFT-DIDACT: in quest'ultimo caso, se la situazione problematica non è già emersa, si rende necessaria la riformulazione del software.

Qualora, invece, le valutazioni collettive si presentassero generalmente positive, ma si riscontrassero frequenti risposte errate a livello individuale, il problema dovrà indirizzarsi all'analisi delle capacità di apprendimento del singolo alunno. In tal caso si rende necessaria una ristrutturazione semplificata del software e, mantenendo intatta l'organizzazione DIDACT e QUESTY, vanno ricercate le unità didattiche col maggior grado di coinvolgimento software e formulate le domande inerenti l'UD chiave-primaria e/o le domande con minor grado di difficoltà.

7 - CONSIDERAZIONI CONCLUSIVE

La complessità del progetto emerge soprattutto dalla costruzione della macrostruttura vuota le cui caratteristiche essenziali risiedono nell'elasticità dei legami logici, nella possibilità di ulteriori implementazioni, nella rapidità di accesso e nella "facilità" di utilizzo.

Gli operatori pedagogisti, di controllo i progetti corso'. Il compito arduo se di gran lunga tempo e tra unità didattiche verranno violate le gi

Nel nostro ometto si è sempre articolare e distribuzione, considerando a umane, non software di sola scuola

Si rende rete di più canto al ne una scuola li, non un (promuove fronto delle dotto, perm l'accesso, distribuzione dai monop editori.

NOTE:

- 1) Nell'espos ware didattic riale struttur informatici a
- 2) J. P. GOF dattico di un matici e cibe 1966.

Gli operatori scolastici (docenti, esperti, pedagogisti, psicologi, ecc.) dovranno 'riempire di contenuti' tale struttura: saranno loro i progettisti della 'struttura logica del corso'. Il compito si rivelerà immediatamente arduo se affrontato nella sua totalità, ma di gran lunga più semplice se frazionato nel tempo e tra molteplici operatori: alle poche unità didattiche inizialmente immesse ne verranno via via aggiunte di nuove e/o articolate le già esistenti.

Nel nostro progetto abbiamo volutamente ommesso la progettazione hardware; va comunque sottolineata la necessità di articolare razionalmente l'acquisto e la distribuzione degli strumenti informatici. Considerando anche i gravosi costi delle risorse umane, non è più pensabile la fruizione di software didattico ad uso esclusivo di una sola scuola.

Si rende indispensabile l'allacciamento in rete di più istituzioni scolastiche dove, accanto al notevole risparmio hardware (in una scuola sono sufficienti alcuni terminali, non un grande elaboratore centrale), si promuove lo scambio informativo nel confronto delle esperienze e del software prodotto, permettendo nel contempo, non solo l'accesso, ma anche la produzione e la distribuzione di software didattico, svincolati dai monopoli delle case costruttrici e degli editori.

NOTE:

1) Nell'esposizione del nostro progetto il termine *software didattico* viene usato quale sinonimo di 'materiale strutturato e organizzato secondo i principi informatici ad uso didattico'.

2) J. P. GORJUNOV, *Struttura logica e algoritmo didattico di un corso*, in L. B. ITELSON, *Metodi matematici e cibernetici in pedagogia*, Feltrinelli, Milano, 1966.

Fiorino Tessaro, laureato in Psicologia ed in Pedagogia, opera presso il Gruppo di Informatica del Provveditorato agli Studi di Padova in qualità di analista-programmatore.