

CONSIGLIO
NAZIONALE
DELLE RICERCHE

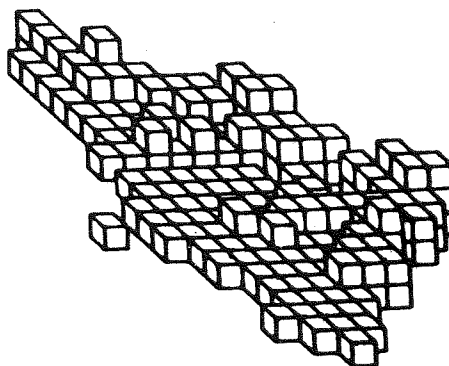
PROGETTO
FINALIZZATO
INFORMATICA

INFORMATIZZAZIONE
DELLA PUBBLICA
AMMINISTRAZIONE

TERRITORIO

**Un sistema di elaborazione e gestione
di carte tematiche**

**B. Biagi
E. Faccioli
M. Morandi Cecchi**



PROGETTO FINALIZZATO INFORMATICA
Sottoprogetto P2: Informatizzazione della Pubblica Amministrazione
OBIETTIVO TERRITORIO

UN SISTEMA DI ELABORAZIONE E GESTIONE DI CARTE TEMATICHE

B. Biagi+, E. Faccioli, M. Morandi Cecchi ++.

1. Si è affrontato il problema della progettazione e implementazione di un package grafico, per l'effettuazione delle più elementari operazioni sulle carte tematiche e sulle relative legende:

- stampa delle informazioni generali del sistema;
- plottaggio delle carte memorizzate;
- stampa delle legende;
- cancellazione di carte già memorizzate;
- modifica dei dati;
- intersezione tra carte.

Quello che si è cercato di ottenere è un sistema economico modulare ed efficiente, che possa facilmente essere usato anche da utenti con conoscenze informatiche molto limitate.

Infine, pensando ad un possibile utilizzo del sistema anche su mini-elaboratori, si è vincolata l'implementazione all'utilizzo di un linguaggio di grande diffusione su questo tipo di computer, il Pascal.

2. I problemi considerati

In relazione agli obiettivi del nostro lavoro, si vogliono descrivere ora i problemi incontrati nella fase di progettazione, le scelte operate e le motivazioni che ci hanno

+ Istituto di Elaborazione dell'Informazione - C.N.R. -
Pisa.

++ Istituto di Matematica, Università degli Studi di Catania.

Lavoro svolto nell'ambito del Progetto Finalizzato Informatica, Sottoprogetto P2, Obiettivo Territorio.

spinto all'utilizzazione di particolari strutture anziché di altre.

Per l'organizzazione del lavoro è stata adottata una politica di compromesso che ha coinvolto parametri quali lo spazio di memoria occupato e l'efficienza del sistema.

Con questi presupposti sono state affrontate tutte le scelte inerenti sia la memorizzazione e l'elaborazione dei dati cartografici, che la determinazione del linguaggio di programmazione da utilizzare come DML.

2.2 Obiettivi del sistema

Nell'ambito della progettazione di un sistema di gestione del territorio regionale, abbiamo incentrato la nostra attività nella realizzazione della parte concernente la memorizzazione, l'elaborazione, la stampa e il plottaggio di dati cartografici.

Trascurando tutti i problemi relativi alla privacy e alla sicurezza dei dati, per cui sono stati previsti solo dei controlli piuttosto blandi, abbiamo ristretto i nostri obiettivi alla costruzione dell'interfaccia tra sistema e utente e all'implementazione di quella parte che concerne il soddisfacimento delle sue richieste.

Come mostrato in fig. 2.1. si sono predisposti dei moduli per l'acquisizione, la restituzione e l'elaborazione dei dati, che dovranno essere strutturati e organizzati in modo compatibile con le esigenze del sistema.

Nostri compiti saranno allora:

- gestione delle aree, punti e attributi relativi alle carte tematiche di base inserite nel sistema;

- integrazione di dati cartografici e numerici;

- elaborazione in maniera interattiva dei dati gestiti secondo le richieste di un utente tipicamente non informatico.

Si fa notare che, nonostante si utilizzi la stessa terminologia dei data-bases, abbiamo voluto realizzare unica -

mente il sistema di gestione ed elaborazione di carte tematiche descritto in fig. 2.1.

La sua implementazione ha richiesto da una parte un accurato studio di tutto ciò che concerne la cartografia automatica e dell'altra la consultazione delle teorie riguardanti l'organizzazione e l'integrazione di dati disomogenei.

Si fa inoltre notare che la trattazione più dettagliata di tutti gli aspetti del nostro sistema grafico sarà fatta man mano che procederemo nella descrizione del lavoro di progettazione.

2.3 Organizzazione e strutturazione dei dati

Primo passo per l'attuazione dei nostri scopi è la stesura di uno schema generale di organizzazione avente tutte le caratteristiche desiderate e strutturato in conformità con l'esigenze di un ipotetico utente.

Bisognerà cercare cioè di integrare i dati in modo che questi siano da una parte, il meno ridondanti possibile e, dall'altra, si prestino facilmente ad una certa attività di elaborazione che si intende prevedere.

Tipicamente avremo a che fare con dati disomogenei riguardanti carte e legende.

Inoltre per velocizzare il lavoro di recupero e di elaborazione, si dovranno inserire delle informazioni aggiuntive in quantità strettamente dipendente dalla complessità delle operazioni consentite.

Tali dati saranno invisibili all'utente, cui sarà presentata una visione logica concernente unicamente zone, carte e relative legende.

Dal momento che non sono state poste delle limitazioni di sorta per quel che riguarda la privatezza dei dati inseriti nel sistema, chiunque ha la possibilità di accedere alla totalità delle informazioni.

Possiamo quindi limitarci alla descrizione dello schema interno e dello schema concettuale in quanto quest'ultimo coincide perfettamente con l'unica visione utente adottata.

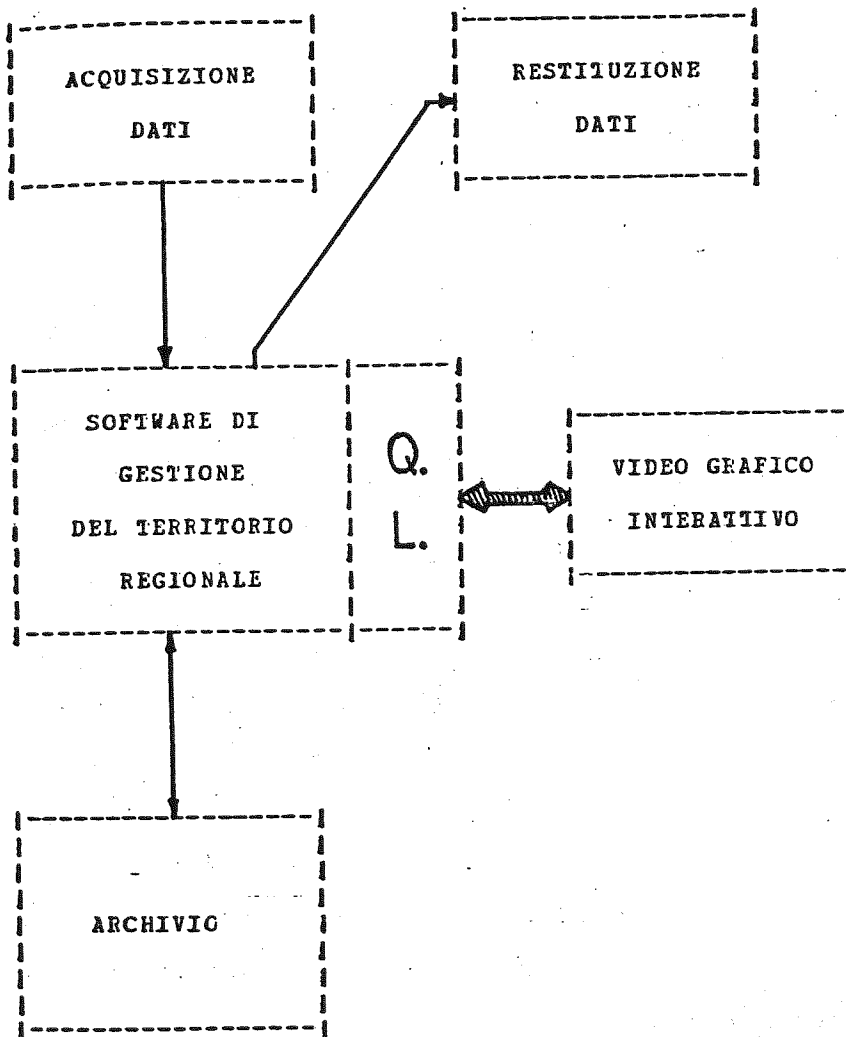


figura 2.1.

Prima di affrontare questo problema ci sembra importante, per delimitare ulteriormente i nostri scopi, dare una somma descrizione delle operazioni che ci sembrano fondamentali per qualsiasi lavoro su dati cartografici.

2.3.1. Il linguaggio di interrogazione

Tipicamente, avendo a che fare con un insieme di carte topografiche, le operazioni standard riguardano la lettura delle informazioni in questo contenute. Fondamentali sembrano a questo punto le legende, che rappresentano la chiave di lettura per le suddette informazioni.

Trattandosi di carte tematiche, inoltre, le legende operano una suddivisione delle carte in classi di equivalenza, raggruppando tutte le aree con una singola proprietà all'interno della stessa classe e descrivendone le caratteristiche.

Volendo garantire il soddisfacimento di gran parte delle richieste possibili, sarà necessario prevedere, oltre alle normali operazioni di inserzione, cancellazione e modifica dei dati, anche quelle di stampa sia delle carte che delle relative legende.

Si è pensato inoltre, di facilitare i compiti dell'utente in due modi:

- il primo gli permette di conoscere i nomi di tutte le zone per cui esiste nel sistema almeno una carta;
- il secondo invece fornisce il codice, il tipo e la scala di tutte le carte relative ad una singola zona.

Rimandiamo per ora la trattazione del modo in cui saranno memorizzate le carte, accennando unicamente al fatto che ogni carta è individuata univocamente all'interno del sistema dal suo codice. La conoscenza di tali codici, resa possibile mediante la seconda facilitazione, permette all'utente di accedere a tutte le carte di suo interesse, mediante la selezione di un opportuno comando.

Si è voluta dare, inoltre, la possibilità di costruire carte tematiche di sintesi ottenute da quelle di base mediante l'operazione di intersezione.

In questo modo, avendo a disposizione le carte di fig.2.2. e 2.3., si è in grado di riprodurre una qualsiasi di quelle rappresentate nelle figure 2.4, 2.5., 2.6., 2.7., o un'altra costituita dall'unione di più classi di intersezione (p.e. una carta in cui siano rappresentate sia le classi di fig. 2.4. che quelle di fig. 2.5.).

In più, è stata garantita anche la possibilità di restringere una carta alle sole classi di interesse. Così per esempio dalla carta di figura 2.8. possiamo ottenere il plottaggio di quelle in figura 2.9., 2.10..

Riassumendo, possiamo concludere raggruppando le operazioni che si vogliono gestire nelle seguenti categorie:

- consultazione delle informazioni presenti nel sistema;
- inserzione dati;
- cancellazione dati;
- modifiche dati;
- plottaggio carte;
- stampa legende;
- intersezione carte.

2.3.2. I dati.

Come già accennato, dovendo garantire l'accesso al sistema ad utenti poco esperti, si è resa indispensabile una visione utente estremamente semplice, che, grazie all'ausilio di un linguaggio di interrogazione, garantisca varie facilitazioni per l'utilizzo delle informazioni memorizzate.

2.3.2.1. Organizzazione logica

Daremo ora una descrizione piuttosto sommaria della visione dei dati utente, rimandando la trattazione più accurata al capitolo descrivente il linguaggio di interrogazione. Come schematizzato in fig. 2.11 ciò che l'utente vede riguarda unicamente le carte e le relative legende.

Fra questi tipi di dati esiste una corrispondenza uno-uno che è facilmente gestibile con un sistema gerarchico.

L'accesso alla legenda, infatti è logico ed è selezionato grazie al codice che identifica univocamente la carta al-

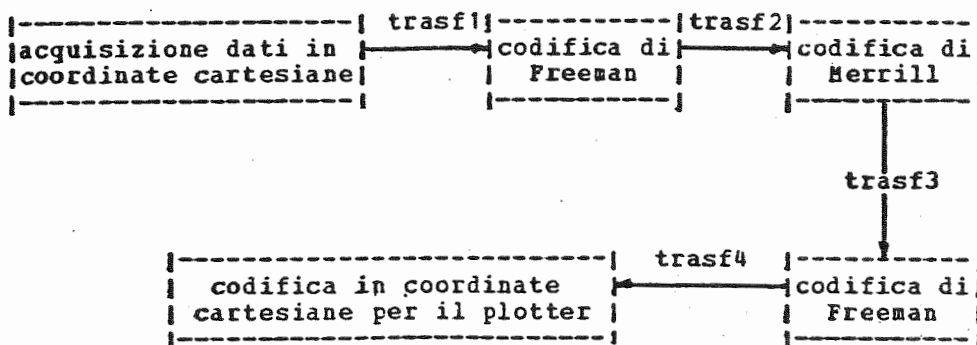


figura 2.2.

METODO	TEMPO DI ELABORAZIONE PER LA FREDISPOSIZIONE AL PLOTTAGGIO DI UNA CARTA DI INTERSEZIONE	
coord. cartesiane	T_{e0}	(2.3.1.)
Freeman	$T_{e1} + T_{f2}$	(2.3.2.)
Merrill	$T_{e2} + T_{m1} + T_{m2} + T_{f2}$	(2.3.3.)

figura 2.3.

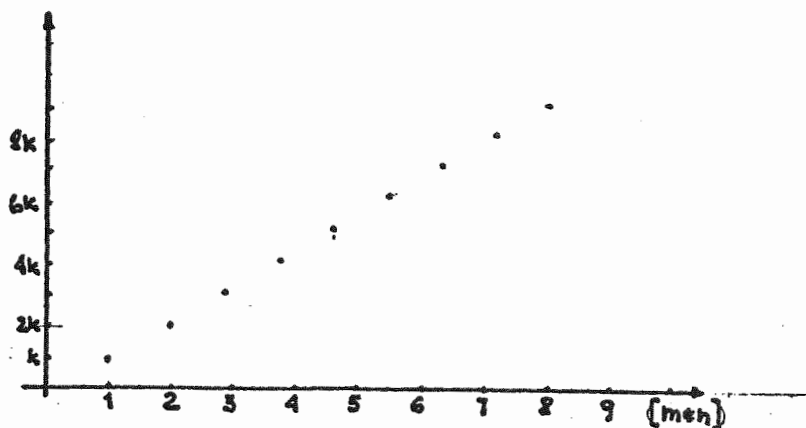


figura 2.4.

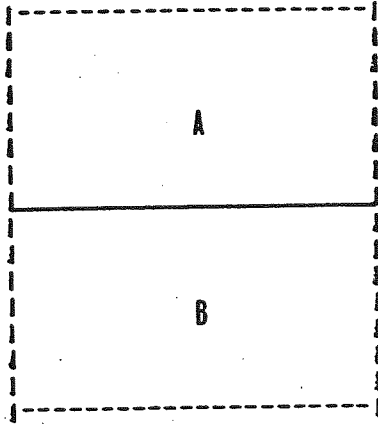


figura 2.5.

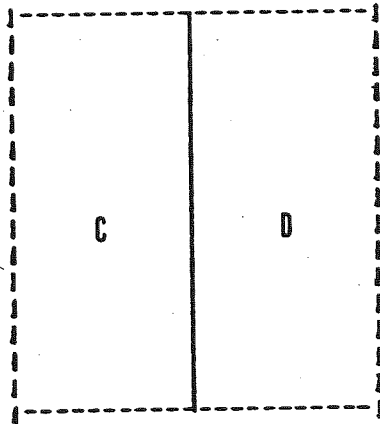


figura 2.6.

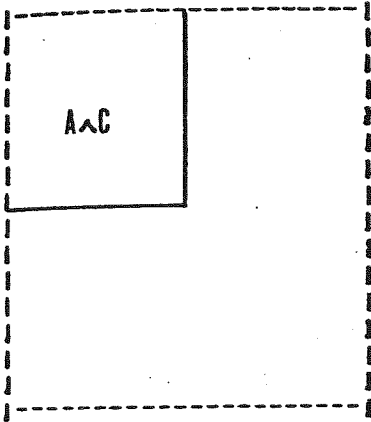


figura 2.7.

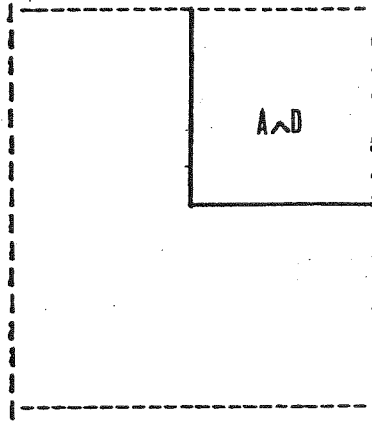


figura 2.8.

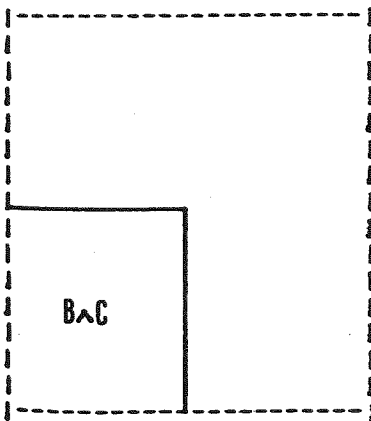


figura 2.9.

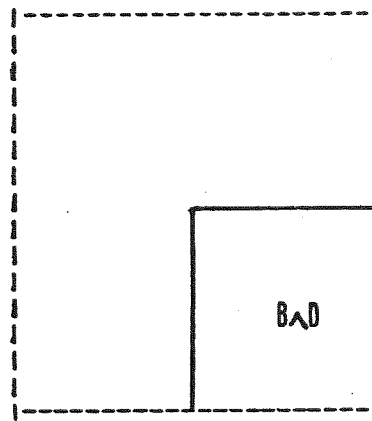


figura 2.10.

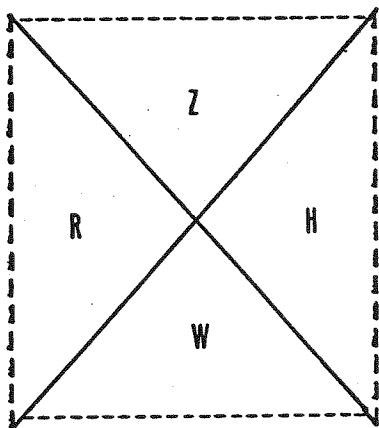


figura 2.11.

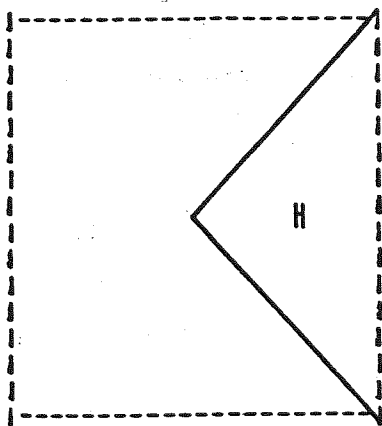


figura 2.12.

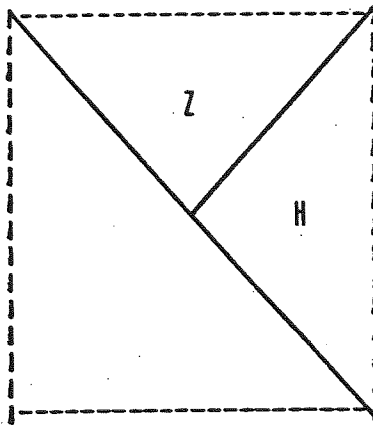


figura 2.13.

l'interno del sistema.

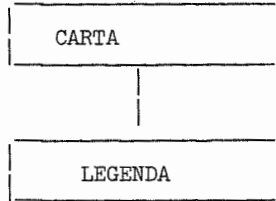


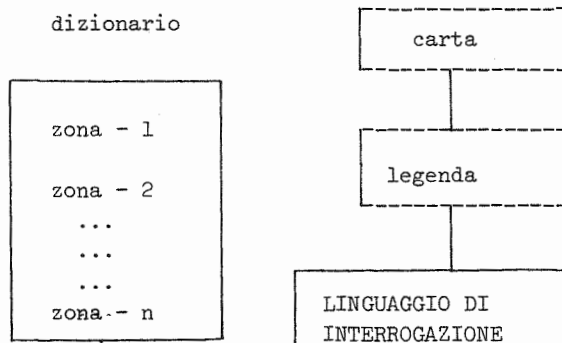
fig. 2.14.

Oltre a ciò, per facilitare un qualsiasi approccio al sistema si è pensato ad una suddivisione logica delle carte a seconda delle zone che queste descrivono.

Tale operazione consente ad un utente di ricevere informazioni solo su quella parte dei dati concernente i suoi interessi.

La fig. 2.12. descrive lo schema logico generale comprendente sia i dati che il dizionario in cui sono inserite tutte le informazioni riguardanti le zone.

Facciamo notare che, dal punto di vista logico, non esiste alcun collegamento diretto tra il dizionario e le carte, in quanto servono a soddisfare diverse esigenze dell'utente.



2.3.2.2. Organizzazione fisica

Data la varietà delle operazioni che si vogliono prevedere è indispensabile organizzare i dati in modo che questi non siano orientati alla singola applicazione ma permetta no, con la minima ridondanza, la gestione e il soddisfacimento di tutte le possibili richieste dell'utente.

Tratteremo in dettaglio i dati riguardanti le carte, quelli concernenti il dizionario, nonché vedremo, in relazione al linguaggio di interrogazione, la convenienza di adottare una soluzione anziché un'altra.

Dal momento che siamo dipendenti dal sistema operativo per quanto riguarda la definizione e la gestione dei files in memoria di massa, sapendo di avere a che fare con una quantità di dati piuttosto notevole, abbiamo voluto spezzare le informazioni riguardanti una sola carta in tre strutture distinte:

- file "carta";
- file "legenda";
- file "attributi".

Nel primo sono inseriti, come mostrato in figura 2.13., tutti i dati riguardanti i tratti componenti una carta. Dovendo gestire anche delle operazioni piuttosto complesse, quali l'intersezione, abbiamo preferito utilizzare la codifica per coordinate cartesiane, nonostante gli svantaggi che questa comporta in termini di spazio di memoria occupato.

file "carta

n-tratti (k)			
1	n-punti	x(1,1)	y(1,1) ...
2		
.....			
k	n-punti	x(k,1)	y(k,1) ...

fig. 2.13

Inoltre, essendo i files CMS sequenziali, si è pensato di memorizzare più punti in ogni record in modo da accedere più velocemente anche agli ultimi tratti. Nel file "legenda" sono invece memorizzati, come schematizzato in fig. 2.14, i dati che consentono la corretta interpretazione dei segni di una carta.

file "legenda"

```
codice-carta  nome-zona  n-classi (k)
1 segno-classe-1  spiegazione-proprietà
2 segno-classe-2  spiegazione-proprietà
.....
.....
k segno-classe-k  spiegazione-proprietà
scala spig(1) spig(2) spig(3) spig(4)
```

fig. 2.14

Notiamo inoltre, che i tratti relativi ad ogni area sono memorizzati per indirizzi crescenti in modo da accedere ai suddetti tratti in un'unica scansione del file "carta". Grazie all'utilizzo di queste informazioni, il sistema è in grado di reperire velocemente tutto ciò che gli serve sapendo esattamente dove trovarlo.

In conformità con le modalità fornite dal CMS, tali files sono identificati all'interno del sistema operativo da un nome, un tipo e un modo. Quest'ultimo seleziona il mini disco sul quale sono memorizzati e dovrà quindi essere uguale per ogni carta. Per identificarla restano quindi disponibili il nome e il tipo: volendo utilizzare quest'ultimo per segnalare le operazioni effettuate di seduta in seduta su ogni carta si è utilizzato a tale scopo il nome. Abbiamo identificato inoltre i files "legenda" e "attributi" aggiungendo al suddetto nome rispettivamente i prefissi "LG" e "AT".

Infine, per quel che riguarda il dizionario, si è pensato di organizzarlo come descritto in fig. 2.16. Volendo inoltre permettere ad un utente di costruirsi dei dati temporanei si sono volute memorizzare le informazioni che lo riguardano non nel dizionario ma a parte in un suo prolungamento valevole solo per seduta in corso. Tale file, infatti, è organizzato fisicamente come il "dizionario", ma è cancellato all'atto dell'uscita dal sistema insieme a tutte le carte "temporaneamente" presenti.

file "dizionario"

N-ZONE			
nome-zona-1	n-carte		
1 codice-carta-1	spiega-carta	scala	
2			
.....			
nome-zona-2	n-carte		
1 codice-carta-1	spiega-carta	scala	
.....			
.....			

fig. 2.16

Nell'ultimo di questi files, che chiameremo "attributi", sono invece inserite, secondo il formato di fig. 2.15., tutte le informazioni necessarie all'attività elaborativa del sistema.

file "attributi"

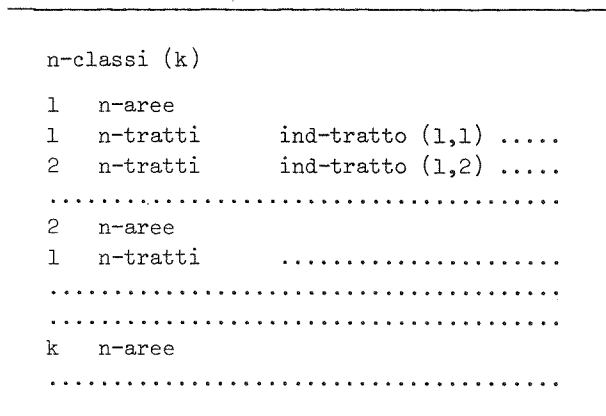


fig. 2.15

3. Il linguaggio di interrogazione

3.1 Introduzione

Per consentire l'interazione col sistema è stato implementato un linguaggio di interrogazione che ha il compito di interpretare le richieste dell'utente man mano che gli vengono fornite.

Tale linguaggio è strutturato gerarchicamente in due livelli, nel primo dei quali viene semplicemente ristretto l'insieme delle possibili richieste, mentre, nel secondo, interattivamente, si ha la completa specificazione dei bisogni informativi dell'utente interpretabili in modo univoco dal sistema.

Tale metodologia di utilizzo può comunque essere accelerata fornendo in un unico comando la totale specificazione delle informazioni desiderate.

Con un'organizzazione di questo tipo si sono voluti aiutare gli utenti poco esperti nella formulazione delle loro

richieste, senza rallentare però il lavoro di coloro che già conoscevano il linguaggio.

3.2 Comandi del linguaggio

I comandi a disposizione dell'utente per interagire col sistema sono raggruppabili in tre categorie:

a) comandi di controllo: sono quei comandi legati all'accesso e all'uscita del sistema:

- Gestfi inizia la sessione e fa entrare nel sistema;
- Esc termina la sessione e fa uscire dal sistema;

b) Comandi per il recupero dei dati: sono i comandi che permettono esclusivamente il recupero delle informazioni presenti nel sistema senza però modificare in alcun modo i dati esistenti:

- Inf fornisce informazioni generali sulle e/o carte memorizzate nel sistema;
- Leg recupera e stampa a terminale la legenda della carta selezionata;
- Spt plotta una carta e la relativa legenda;

c) comandi per la manipolazione dei dati: sono quelli che permettono di modificare l'insieme dei dati nel sistema, inserendo, cancellando, modificando ed elaborando i dati:

- Ins inserisce una nuova carta con la relativa legenda;
- Mod modifica nomi e dati precedentemente già memorizzati;
- Can cancella nomi e/o dati già memorizzati;
- Int produce la carta risultante dall'operazione di intersezione di una o più classi di due carte selezionate e la inserisce poi tra i dati presenti nel sistema.

Eseguito correttamente il comando il sistema è pronto a ricevere il successivo, tranne naturalmente nel caso in cui l'ultimo comando sia stato Esc.

Nell'esaminare i vari comandi utilizzeremo le seguenti convenzioni:

- le parole chiave sono in maiuscolo;
- le parole in minuscolo devono essere sostituite da opportuni valori;
- i codici incolonnati rappresentano l'insieme delle possibilità fornite ad un qualsiasi utente a quel livello di esecuzione del comando;
- i codici fra parentesi tonde sono opzionali.

3.2.1 Il comando GESTFI

Tale comando permette l'accesso al sistema e fornisce all'utente la possibilità di utilizzare uno dei comandi del linguaggio di interrogazione.

Il suo formato é:

GESTFI	
--------	--

L'accesso è garantito dalla stampa di tutti i codici dei possibili comandi.

3.2.2 Il comando ESC

E' il comando complementare al precedente e serve appunto per uscire dal sistema.

Il suo formato é

ESC	
-----	--

L'uscita dal sistema è completata dalla stampa del messaggio di ritorno di un qualsiasi comando del CMS (R_i).

3.2.3 Il comando INF

E' il comando che permette di conoscere in ogni istante le reali possibilità informative del sistema.

Il suo formato è:

INF	ZONE nome-zona
-----	-----------------------

dove: nome-zona è il nome della zona che interessa l'utente.

Le funzioni svolte sono:

- ZONE: produce la stampa a terminale di tutte le zone per le quali è presente nel sistema almeno una carta;
- nome-zona: visualizza per ogni carta riguardante la zona selezionata il relativo codice, il tipo e la scala;

Nel caso vengano richieste informazioni su una zona non presente nel sistema, il comando opera come se fosse stato selezionato il comando "INF ZONE".

3.2.4 Il comando LEG

Fornisce la stampa a terminale della legenda della carta selezionata.

Il suo formato è:

LEG	codice-carta
-----	--------------

dove: codice-carta è il codice che permette di identifi
care univocamente una carta e la relativa legenda.

3.2.5 Il comando SPT

E' il comando che permette il plottaggio di una carta o
solamente di alcune sue classi.

Il suo formato è:

SPT	codice-carta (CLASS k)
-----	------------------------

dove: -k è una sequenza non vuota di indici di classi pre
scelte per il plottaggio.

Nel caso non venga fornita la specifica della seconda op
zione verrà plottata l'intera carta.

3.2.6 Il comando INS

E' l'unico comando che permette l'inserzione di una carta
tematica di base nel sistema.

Il suo formato é:

INS	nome-zona codice-carta	T
		f1 f2

dove: f1, f2 sono i nomi, i tipi e i modi dei files CMS
in cui sono memorizzati, secondo un'opportu

no formato, rispettivamente i dati riguardanti la carta e la legenda.

Nella costruzione di tali files bisognerà attenersi ai formati descritti nelle fig. 3.1 e 3.2.

Valutiamo ora le funzioni eseguite:

- T avverte che la carta da inserire è già presente fra i dati temporaneamente nel sistema (dati che saranno persi alla fine della corrente seduta a terminale); la memorizzazione, in questo caso, consiste nell'inserimento di tale carta fra i dati permanenti del sistema.
- fl f2 permettono l'inserzione permanente di una nuova carta che dovrà essere predisposta nel file 'fl' come in fig. 3.1 e nel file 'f2' come in fig. 3.2. In entrambe tali files è di fondamentale importanza unicamente attenersi alla sequenza di memorizzazione dei vari dati; non è assolutamente determinante, invece, la spaziatura fra questi ultimi (per esempio le coordinate di un tratto possono essere memorizzate in quantità diversa da quella indicata in fig. 3.1 e 3.2.

Formato del file fl:

n						
1	p1	x(1,1)	y(1,1)	x(1,p1)	y(1,p1)
2	p2	x(2,1)	y(2,1)	x(2,p2)	y(2,p2)
.....						
.....						
n	pn	x(n,1)	y(n,1)	x(n,pn)	y(n,pn)

n è il numero totale dei tratti della carta;
pi (i = 1,n) è il numero dei punti dell'i-esimo tratto;

$x(i,j)$ $y(i,j)$ sono le coordinate cartesiane del j -esimo punto dell' i -esimo tratto.

Il formato del file f2 è:

```
k
1  a1  s1  n1
1  tr(1,1) t(1)...t(tr(1,1))  2tr(2,1) t(1)...t(tr(2,1)).
2  a2  s2  n2
1  tr(1,2) t(1)...t(tr(1,2))  2tr(2,2) t(1)...t(tr(2,2)).
.....
.....
k  ak  sk  nk
1  tr(1,k) t(1)...t(tr(1,k))  2tr(2,k) t(1)...t(tr(2,k)).
scala x(1) y(1) ..... x(4) y(4)
```

dove: k è il numero totale delle classi in cui è suddivisa la carta;

a_i è il numero dell' i -esima classe (lblank 7char);

s_i è la spiegazione della proprietà dell' i -esima classe (lblank 39char);

n_i è il numero delle aree distinte per cui vale la proprietà dell' i -esima classe;

$tr(i,j)$ è il numero dei tratti che compongono la j -esima area dell' i -esima classe;

$t(i)$ è l' i -esimo indice ditratto dell'area precedentemente riferita;

scala è la scala della carta;

$x(i)$ $y(i)$ ($i = 1,4$) sono le coordinate degli spigoli della carta.

3.2.7 Il comando MOD

Questo comando permette la modifica di alcuni nomi già memorizzati nel sistema.

Il suo formato é:

	CARTA	codice-cartal	codice-carta2
MOD	ZONA	codice-carta	nome-zonal nome-zona2

Esaminiamo le funzioni eseguite:

- CARTA modifica il codice della carta con uno nuovo dopo aver controllato che quest'ultimo non sia già valido per un'altra carta;
- ZONA modifica il nome della zona cui è riferita la carta selezionata con uno nuovo

3.2.8 Il comando CAN

Effettua la cancellazione di dati già memorizzati.

Il suo formato è:

CAN	nome-zona	codice-carta
-----	-----------	--------------

Esaminiamo le funzioni eseguite:

- l'unico tipo di cancellazione previsto riguarda la distruzione totale dei dati concernenti una singola carta e nel caso questa sia la sola riferita ad una particolare zona viene eliminata dal sistema anche quest'ultima.

3.2.9 Il comando INT

Permette la costruzione di una nuova carta risultante dall'intersezione di altre due già presenti nel sistema e relative alla medesima zona.

Il suo formato è:

```
INT  cod-cartal (CLASS k1) CON cod-carta2 (CLASS k2)
```

dove: - k1,k2 sono le sequenze dei numeri d'ordine delle classi selezionate.

Così, per esempio, 'INT cod-cartal CON cod-carta2' produce come risultato una carta derivata dalle due selezionate e rappresentante l'intersezione di ogni classe della prima con ogni classe della seconda, mentre selezionando il comando 'INT cod-cartal CLASS i CON cod-carta2 CLASS j' si ottiene l'intersezione dell'i-esima classe della prima carta con la j-esima della seconda.

Anche in questo caso verrà richiesto all'utente un codice con cui intende individuare la nuova carta, che sarà poi inserita temporaneamente o definitivamente a seconda dei desideri dell'utente tra i dati del sistema.

Si fa notare, inoltre, che volendo costruire la carta risultante dall'operazione di restrizione bisognerà duplicare tale carta e richiedere l'operazione di intersezione tra la carta e la sua copia.

3.3 Osservazioni finali

Per quel che riguarda gli errori, ogni volta che se ne presenta uno, il sistema stampa un opportuno messaggio di spiegazione ed a seconda della gravità della situazione decide se mettersi in attesa di un nuovo comando o uscire dal sistema.

In ogni caso il riconoscimento di un errore provoca la non-esecuzione del comando selezionato.

Ogni volta che il sistema è in attesa di un comando, la

selezione del codice '?' provoca la stampa a terminale di una spiegazione dei comandi consentiti.

Tutte le situazioni non previste nella precedente trattazione vengono sempre risolte con una interazione tra sistema e utente mediante scambio di messaggi.

BIBLIOGRAFIA

- Albano 80 Albano A., Orsini R., Progetto di basi di dati, Opera Universitaria, Pisa, 1980.
- Bracchi 79 Bracchi G., Martella G., Pelagatti G., Sistemi per la Gestione di Basi di Dati.
- Biagi 80 Biagi B., Montani C., Un Algoritmo generale per determinare le regioni di intersezione tra con torni poligonali, Nota interna C.N.R., Pisa, 1980.
- Montani 81 Montani C., Morandi Cecchi M., Pazzini F., Castini P. Una Metodologia automatica per la generazione di carte tematiche in sintesi, Rapporto dell'Obiettivo Territorio P.F. Informatica, Sottoprogetto P2.