

INDICE

1. Introduzione.....	2
2. La raccolta dei dati.....	4
2.1 Database della scuola.....	4
2.2 Questionario da compilare.....	4
2.3 Interfaccia Web.....	5
3. La normalizzazione degli indirizzi.....	6
3.1 In che cosa consiste e perché normalizzare.....	6
3.2 Descrizione del problema affrontato e possibile soluzione.....	6
3.2.1 Il viario della città.....	6
3.2.2 Le funzione di somiglianza tra stringhe.....	6
3.2.3 Funzione di estrazione della DUG.....	8
3.2.4 Funzione di estrazione del numero civico.....	8
3.3 Cenni ad altri possibili algoritmi.....	9
4. Il sistema sviluppato.....	10
4.1 Il sistema GIS.....	10
4.2 La struttura generale.....	12
4.3 Il database alfanumerico.....	14
4.4 Utilizzo di ODBC e VisualBasic come integratori di ambienti.....	15
5. I dati territoriali.....	17
5.1 Banca dati geografica.....	17
5.2 La zonizzazione: Zone Istat, Zone di traffico e Macrozone.....	17
5.3 Trasferimento dei dati relativi alle scuole sul sistema GIS.....	20
6. La mobilità del comune di Livorno.....	20
6.1 La creazione della matrice O/D e delle linee di desiderio.....	20
6.2 Conclusioni.....	25
7. Riferimenti.....	27

1. INTRODUZIONE

Il progetto prende il nome dall'espressione "Linee di desiderio": in un'analisi di tipo trasportistico, una linea di desiderio è una linea astratta relativa alla distanza in linea d'aria tra due punti di riferimento: il baricentro di una zona origine ed il baricentro di una zona destinazione. Linee di questo tipo, utilizzate in un contesto territoriale definito, permettono la rappresentazione della distribuzione spaziale della domanda di mobilità di utenti che si spostano sulla base di una matrice origine-destinazione (OD) in cui:

- ogni colonna rappresenta un punto di origine;
- ogni riga rappresenta un punto di destinazione;
- il valore dei singoli elementi rappresenta il numero di spostamenti (domanda di mobilità) relativi alle varie coppie OD; ognuno di essi viene rappresentato graficamente dallo spessore di una specifica linea di desiderio.

In genere il riferimento temporale di una matrice OD è piuttosto contenuto e, come nel nostro caso, si riferisce ad una fascia oraria ben precisa (es. 7.30-8.30) dove gli spostamenti per il fenomeno studiato sono particolarmente significativi.

Scopo del lavoro è stato quello di *definire una metodologia da porre in atto per creare un sistema informativo per analizzare la domanda di mobilità relativa agli studenti delle scuole superiori della città di Livorno che nella mattina confluiscono nel polo scolastico sull'asse Via Galilei - Piazza Il Giugno*. I risultati ottenuti potrebbero essere usati dalle autorità competenti (nel nostro case-study, l'azienda ATL e le amministrazioni Comunale e Provinciale di Livorno) per prendere delle decisioni al fine di operare scelte nella pianificazione del servizio di trasporto pubblico urbano (ed extraurbano). L'idea di fondo è quella di offrire all'utenza una qualità dei servizi che invogli i cittadini a preferire il trasporto pubblico rispetto a quello privato verso una mobilità sostenibile che miri a ridurre il traffico ed alcuni fenomeni connessi sia da un punto di vista ambientale (emissioni inquinanti - ad esempio CO, NOx, polveri sottili, emissioni acustiche, ecc.) che energetico (consumi di carburante quali benzina, gasolio, GPL).

Il lavoro è articolato su diversi punti descritti nel presente documento:

- raccolta dei dati relativi agli studenti delle scuole di Livorno dell'asse Via Galilei - Piazza Il Giugno
- normalizzazione degli indirizzi di abitazione
- zonizzazione del territorio
- rappresentazione grafica dei dati mediante un sistema GIS
- creazione e analisi delle linee di desiderio

Lo scopo di fondo del lavoro è anche quello di illustrare in linea di principio i passaggi fondamentali che si effettuano nella progettazione di uno specifico DSS (Decision Support System), ovvero un sistema software di supporto alle decisioni. Un DSS permette di aumentare l'efficacia dell'analisi dei dati relativi a specifici fenomeni in quanto fornisce supporto a tutti coloro che devono prendere decisioni strategiche di fronte a problemi che non possono essere risolti solo con i modelli della ricerca operativa. Una delle sue funzioni principali è quella di estrarre in poco tempo e in modo versatile le informazioni utili ai processi decisionali, provenienti da una rilevante quantità di dati.

Gli aspetti essenziali di un DSS si possono riassumere in:

- facilità d'uso alla portata di tutti gli utenti
- ambiente interattivo

- possibilità per il sistema di fornire supporto decisionale
- efficacia dell'utilizzo dei modelli e dell'analisi dei dati

Il loro utilizzo però non è circoscritto nell'ambito di una singola azienda, ma è utilizzato in ogni ambito nel quale sia necessario ottenere, da una notevole quantità di dati di cui si è in possesso, delle informazioni e delle conoscenze significative utili alla strategia decisionale.

Esempi di applicazioni possono essere:

- nel trasporto pubblico e privato: pianificazione dei flussi veicolari (piani urbani del traffico)
- nel commercio: analisi di vendite, reclami, spedizioni, inventari, clienti, rimanenze.
- nel settore manifatturiero: analisi di costi di produzione, fornitori, ordini.
- nella sanità: analisi di ricoveri, dimissioni, contabilità.
- nelle telecomunicazioni: analisi del flusso delle chiamate, profilo dei clienti, assistenza clienti, promozioni.
- nel settore della logistica dei trasporti: gestione parco mezzi, le gestione del carico e la distribuzione.

Un DSS aumenta soprattutto l'efficacia delle decisioni cognitive e questo è un fatto rivoluzionario, perché l'obiettivo dell'informatica negli ultimi 30 anni è stato l'aumento dell'efficienza e un intervento "intelligente" nella risoluzione di problemi.

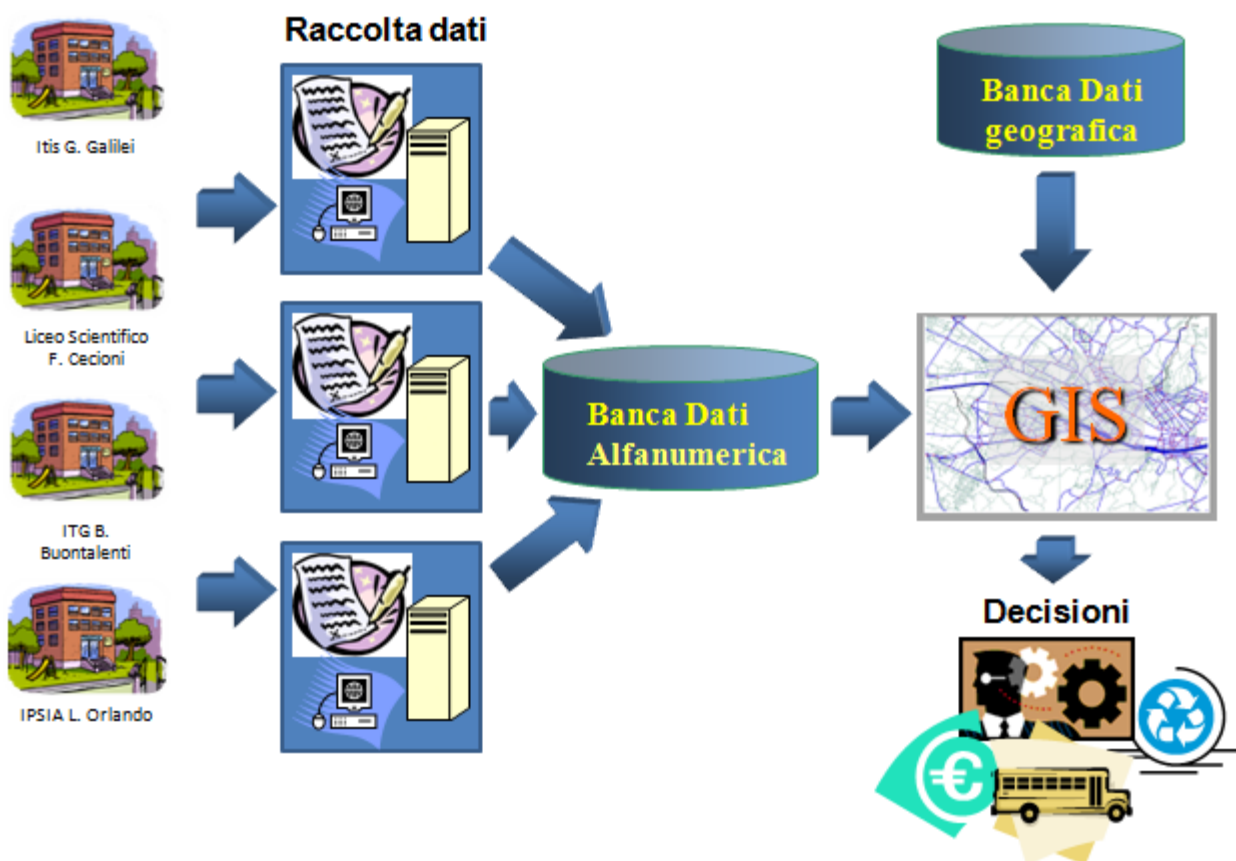


Figura 1 Le varie fasi del progetto

2. LA RACCOLTA DEI DATI

La prima operazione di tutto il lavoro da svolgere è la raccolta dei dati relativi al fenomeno che si vuole studiare. Nel nostro caso si vogliono raccogliere i dati relativi agli studenti delle varie scuole di Livorno, che possono essere ad esempio la scuola che si frequenta, la classe, il sesso, il CAP, il mezzo di trasporto utilizzato per lo spostamento, e necessariamente la città e la via di residenza. Tutti questi dati poi verranno trasferiti su un'opportuna mappa, in modo da tracciare le varie direttrici degli spostamenti all'interno della città.

Il metodo per la raccolta dei dati deve essere quanto più efficiente possibile, ovvero deve essere veloce e non deve produrre dati non coerenti. I metodi possono essere diversi, ed i più comuni, in generale, sono tre.

▪ 2.1 Database interno alla scuola

Il metodo più immediato per la raccolta dei dati è indubbiamente richiedere alla scuola il database contenente l'elenco di tutti gli alunni iscritti. Con questa soluzione non ci sono particolari svantaggi, in quanto la velocità con la quale si ottengono le informazioni è notevole, ed è identificabile con il tempo di risposta da parte della scuola, e non dovrebbe presentare particolari incongruenze nei dati, in quanto essi sono già amministrati dall'organismo di competenza interno alla scuola a cui si è effettuata la richiesta. L'unico svantaggio è che non si possono raccogliere informazioni relative al mezzo di trasporto usato, in quanto questo dato non è contenuto nei database scolastici.

Questo metodo è senza dubbio ottimale, ma non è sempre detto che le scuole siano favorevoli al cedimento dei propri database.

▪ 2.2 Questionario da compilare

Un secondo modo per la raccolta dei dati è la consegna nell'intera scuola, classe per classe, di un questionario, o modulo, da compilare da ciascun alunno, in modo anonimo. Gli svantaggi per questo metodo sono diversi, ad esempio:

Progetto 2011 - Linee di Desiderio
Modulo per la raccolta dati, da compilare in modo anonimo
Scuola di appartenenza:

<input type="radio"/> ITIS G. Galilei	<input type="radio"/> Liceo F. Cecioni	<input type="radio"/> ITN A. Cappellini
<input type="radio"/> ISIS N. Palli	<input type="radio"/> Liceo F. Enriques	<input type="radio"/> ITC A. Vespucci
<input type="radio"/> IPSIA L. Orlando	<input type="radio"/> Liceo Niccolini e Guerrazzi	<input type="radio"/> ITG B. Buontalenti

Classe Sesso ☐ M ☐ F

Provincia Comune

Via Residenza

Mezzo di trasporto utilizzato:

<input type="radio"/> Autobus	<input type="radio"/> Treno	<input type="radio"/> Piedi
<input type="radio"/> Macchina (Passeggero)	<input type="radio"/> Macchina (Conducente)	<input type="radio"/> Bicicletta
<input type="radio"/> Motorino	<input type="radio"/> Altro (Specificare:)	

Figura 2 Esempio di modulo compilabile

- Raccogliere i dati in forma cartacea vuol dire doverli trasferire su un supporto digitale, su un computer, in modo da poterli utilizzare per i nostri scopi. Se fatta manualmente, questa operazione può richiedere diverso tempo, anche se si può ricorrere ad una soluzione di acquisizione ottica, se si è in possesso dei mezzi appropriati per farla. In entrambi i casi però si va incontro comunque ad errori di trascrizione e di interpretazione dei dati scritti su carta.
- Per la correttezza dei dati raccolti ci si affida direttamente ad ogni singolo alunno, e non ad un'entità superiore come poteva essere nel caso precedente.
- La distribuzione di questionari in un'intera scuola può richiedere un costo organizzativo, in quanto è necessario l'utilizzo di una significativa quantità di carta e di inchiostro. E' necessario quindi scegliere in modo adeguato i prodotti da utilizzare, cercando di contenere anche un eventuale danno ambientale, ad esempio favorendo l'utilizzo di carta riciclata.

▪ 2.3 Interfaccia web

In alternativa, è possibile costruire un'interfaccia web da poter utilizzare per inviare i propri dati. Essa, nel nostro caso, potrebbe essere inclusa, ad esempio, nel sito della scuola, in una opportuna sezione relativa ai Progetti. Se non è possibile una soluzione del genere, si deve provvedere a mettere in Internet la suddetta pagina, in modo che sia raggiungibile da qualsiasi macchina in grado di navigare.

Con questo metodo non si ha il problema della trascrizione dei dati, in quanto essi vengono immessi già in forma digitale. Inoltre possono essere immagazzinati in modo automatico dentro il database contenente tutti i dati, usando un linguaggio di scripting apposito in grado di dialogare con un DBMS, come ad esempio il linguaggio PHP.

Non si presenta nemmeno il problema dell'uso di risorse materiali come la carta, contendendo quindi anche l'eventuale danno ecologico, seppur di piccola intensità. Rimane comunque il

problema dell'affidabilità del dato, in quanto, oltre ad essere nelle stesse condizioni precedenti, si aggiungono altri problemi, derivati dall'inaffidabilità del web: lo spam e il flooding (ovvero l'immissione di una grande quantità di dati non coerenti in un breve lasso di tempo, che possono essere effettuati sia da persone, che da programmi che svolgono appositamente questo compito, tipicamente chiamati *bot*).

Per contrastare queste situazioni, si devono prendere delle precauzioni sul lato del server. Per esempio si potrebbe pensare, prima di sottomettere i dati, di far risolvere all'utente un codice *Captcha* (*Completely Automated Public Turing test to tell Computers and Human Apart*), che, come indica il nome, serve a stabilire tramite un test se l'utente sia effettivamente un umano oppure un computer (un *bot*). Il test captcha più utilizzato in assoluto è quello in cui

si richiede all'utente di scrivere quali

siano le lettere o numeri presenti in una sequenza di lettere o numeri che appaiono distorti od offuscati sullo schermo. I captcha però potrebbero non bastare, quindi si possono aggiungere anche altre tecniche, come il bloccaggio degli indirizzi ip: ogni volta che un utente sottomette i propri dati, si salvano, dentro una tabella di un database, il suo indirizzo ip e l'ora in cui è stata eseguita l'operazione. In base a questo database poi si ispezioneranno le sottomissioni, controllando che l'ip dell'utente che vuole immettere i dati non sia presente tra i record della tabella. È necessario comunque dare una scadenza ai record, mantenendoli nella tabella, ad esempio, per al massimo un giorno.

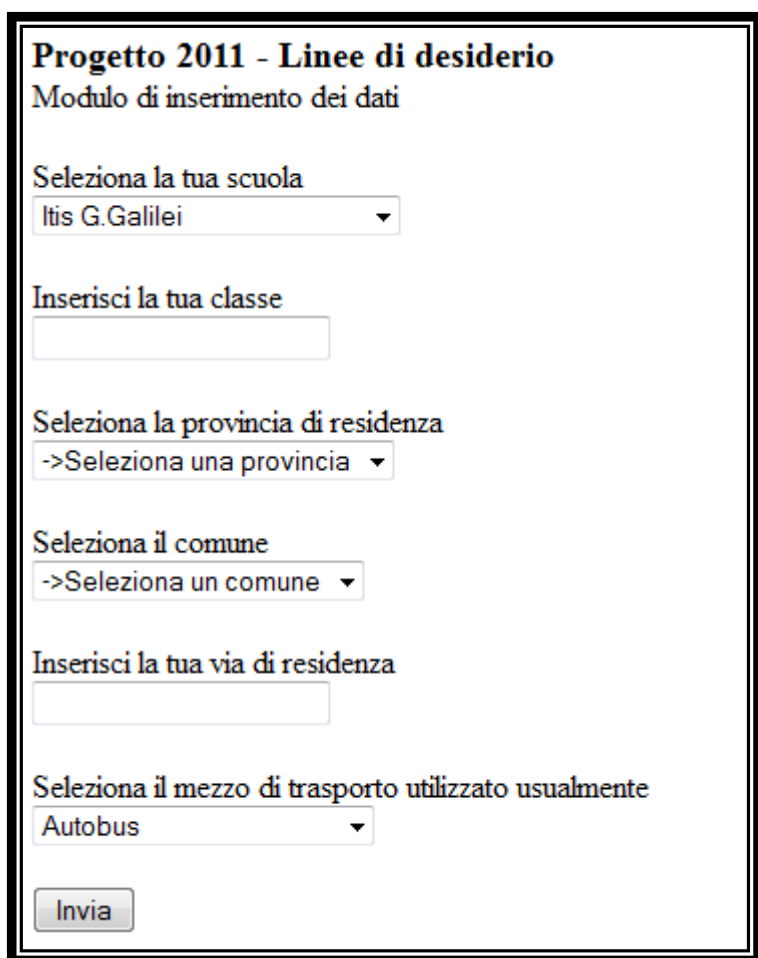


Figura 3 Esempio di interfaccia web

3. LA NORMALIZZAZIONE DEGLI INDIRIZZI

▪ 3.1 In cosa consiste e perché normalizzare

In generale con la parola *Normalizzazione* si intende un qualsiasi tipo di processo che ha l'obiettivo di modificare un oggetto, renderlo più "normale", portandolo ad una situazione più usuale, in modo che non presenti anomalie.

Nel nostro caso l'oggetto da normalizzare è un indirizzo postale. Questo significa: individuare la DUG (Denominazione Urbanistica Generica, ad esempio Piazza, Via, Largo, ecc), correggere errori di scrittura, completare le informazioni mancanti, controllare l'effettiva esistenza del toponimo, individuare in modo corretto il numero civico, ecc.

Il processo di normalizzazione degli indirizzi è un'operazione fondamentale per qualsiasi azienda che voglia assicurarsi il corretto svolgimento di operazioni quali campagne di marketing diretto, promozioni, attività di CRM, spedizioni e molti altri servizi. I problemi di un indirizzo non corretto infatti vengono alla luce quando merci, corrispondenza o mailing sono restituite al mittente, quando cresce il numero dei resi o diminuisce quello degli ordini. Introducendo la normalizzazione, viene garantita la corretta postalizzazione, riducendo di conseguenza le spese postali e pubblicitarie e permettendo l'accesso a tariffe più convenienti.

▪ 3.2 Descrizione del problema affrontato e possibile soluzione

L'obiettivo è quello di normalizzare gli indirizzi di residenza relativi agli studenti delle scuole della città di Livorno, in modo da poterli collocare nella posizione corretta sulla mappa, e poi usare questi dati per analizzare la mobilità all'interno del comune.

C'è stato quindi il bisogno di creare un'applicazione software in grado che faccia corrispondere ad ogni record di ogni database il giusto indirizzo associato. Questo software si basa su alcuni elementi fondamentali: il possesso di un viario della città, contenente tutti i toponimi scritti in modo corretto ed esteso; una funzione che fornisce, in forma probabilistica, la somiglianza tra due stringhe, in modo da ricercare il corretto indirizzo nel viario; una funzione che individua la DUG, separandolo dal resto del toponimo; una funzione che individua il numero civico nell'indirizzo.

Analizziamo ora uno per uno questi elementi.

➤ 3.2.1 Il viario della città

Il viario della città di Livorno consiste in una tabella contenente i toponimi scritti in forma estesa, i toponimi scritti in forma abbreviata e la corrispettiva DUG, senza che contengano anomalie. Nel caso in cui non si riscontri una corrispondenza nella base di dati, l'utente sarà in grado di poter aggiungere il record relativo.

➤ 3.2.2 La funzione di somiglianza tra stringhe

La funzione adottata per stabilire la somiglianza tra le stringhe è definita come una funzione

ricorsiva. Date due stringhe A e B, si cerca la massima sottostringa comune tra di esse e se ne memorizza la lunghezza; a questo punto, in ogni stringa verrà identificata una parte a sinistra della massima sottostringa comune e una a destra; verrà riapplicato quindi l'algoritmo tra le due parti destre e le due parti sinistre, come illustrato in figura. Quando non si trova una massima sottostringa comune,

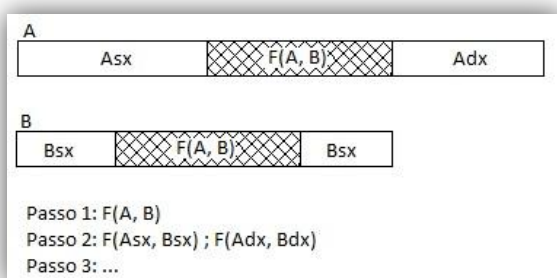


Figura 4 Raffigurazione dell'algoritmo

l'algoritmo è concluso.

Adesso, assumendo che A sia un indirizzo da normalizzare, B un record del viario e sia detto M il numero di caratteri trovati con l'algoritmo precedente, la probabilità di somiglianza tra le due stringhe sarà data dall'espressione:

$$P(A,B) = \frac{1}{2} \left[\frac{2M}{Len(A) + Len(B)} + \frac{M}{Len(A)} \right]$$

Il risultato è quindi dato dalla somma di due quozienti, il tutto diviso per due. Il primo quoziente è dato dal doppio dei caratteri corrispondenti e la somma delle lunghezze delle stringhe: il secondo quoziente è la percentuale di caratteri corrispondenti nell'indirizzo da normalizzare.

In una prima fase di testing dell'algoritmo, applicato senza apportare nessuna modifica agli indirizzi ed assegnando in modo automatico il risultato con probabilità maggiore, è stato riscontrato che circa al 15% dei record non era stato assegnato il corretto toponimo, su un database di 1080 record. Si è pensato allora di eliminare dalle stringhe tutti i segni di punteggiatura (tipicamente apostrofi, virgole e punti), gli spazi e le proposizioni, sia semplici che articolate (ad esempio "DI", "DEGLI", "DELLA", ecc). Applicando queste modifiche alle stringhe ed usando la stessa strategia precedente, la percentuale di errore riscontrata è adesso di circa il 9%, migliorando quindi sensibilmente la situazione. Un altro tentativo per migliorare i risultati è stato quello di modificare ulteriormente le stringhe da andare a confrontare, eliminando anche le vocali. Il risultato non ha ottenuto però non è stato quello aspettato, in quanto la percentuale di errore è salita al 27%. Si è pensato quindi di fare scegliere all'utente i record per i quali non si trova una corrispondenza univoca, suggerendo i risultati con probabilità di somiglianza maggiore del 70%. Con questo metodo l'errore nell'assegnazione è sceso a circa al 3%: questo errore è dovuto in larga parte al fatto che il viario che ci è stato fornito non è aggiornato, e quindi non sono presenti delle vie di recente costruzione.

I risultati ottenuti dai test sono riportati nella seguente tabella:

Metodo Adottato	Record Totali	Corretti	Non Corretti	Non Assegnati	Percentuale errore
Semplice funzione di somiglianza tra i toponimi, dividendo solo la via dal numero civico. Si considera la percentuale di somiglianza più alta. Nel confronto non si leva niente dalle stringhe	1180	995	185	0	15,6779661
Semplice funzione di somiglianza tra i toponimi, dividendo solo la via dal numero civico. Si considera la percentuale di somiglianza più alta. Nel confronto si levano le proposizioni, ma si lasciano le vocali	1180	1065	115	0	9,745762712
Ricerca delle corrispondenze univoche, oppure scelta dell'utente tra i risultati più probabili. Nel confronto si levano le proposizioni, ma si lasciano le vocali	1180	1118	0	62	5,254237288
Semplice funzione di somiglianza tra i toponimi, dividendo solo la via dal numero civico. Si considera la percentuale di somiglianza più alta. Nel confronto si levano le proposizioni e le vocali	1180	861	319	0	27,03389831
Ricerca delle corrispondenze univoche, oppure scelta dell'utente tra i risultati più probabili. Nel confronto si levano le proposizioni, ma si lasciano le vocali (Versione Migliorata)	1180	1135	0	45	3,813559322

➤ 3.2.3 Funzione di estrazione della DUG

Per procedere all'estrazione della DUG (Denominazione Urbanistica Generica), c'è stato bisogno di fare due analisi preventive: una sul viario, trovando tutte le possibili DUG adottate, mentre l'altra per trovare tutte le compressioni. I risultati ottenuti sono contenuti nella seguente tabella:

DUG	ABBREVIAZIONI			
BORGH	B.GO			
BRETELLA				
CORSO	C.SO			
LARGO				
PIAZZA	P.ZZA	P.	P.ZA	
PIAZZALE				
S.P.	PROV.	PROVINCIALE	P.	VIA P.
S.R.				
S.S.				
S.V.				
SCALI				
VIA	V.			
VIALE	V.LE			
VICOLO				

Osservando i risultati trovati possiamo vedere che l'abbreviazione "P." si può riferire sia a "PIAZZA" che a "STRADA PROVINCIALE", quindi non si può fornire un risultato certo per questo tipo di abbreviazione. Si è deciso quindi di non considerare questa compressione e correggere soltanto quelle nelle quali esiste la corrispondenza univoca. Per fare questa operazione, si prende, partendo da sinistra, la stringa contenente l'indirizzo da normalizzare fino al primo spazio, e poi, mediante un sistema di switch, assegnargli la corretta DUG associata.

➤ 3.2.4 Funzione di estrazione del numero civico

Dopo aver estratto la corretta DUG, per normalizzare un indirizzo postale c'è anche il bisogno di dividere il numero civico dalla via di residenza. Per applicare l'algoritmo, è necessario modificare la stringa, sostituendo tutti i segni di punteggiatura (in particolare i punti e le virgole) con uno spazio bianco. Il primo passo poi è quello di cercare tutte le occorrenze di una stringa composta da uno spazio seguito da un numero, salvando la relativa posizione di inizio in un vettore. Dopodiché, in base alle occorrenze trovate, si procede in modi diversi:

- Un'occorrenza trovata: si divide la stringa nella posizione dell'unica occorrenza trovata.
- Più di un'occorrenza: se è stata trovata più di una occorrenza, può darsi che la via di riferimento contenga un numero, come ad esempio una data. Le date non costituiscono un problema, in quanto è sufficiente controllare se è presente il nome di uno dei dodici mesi dell'anno; il problema sono i toponimi che possono contenere dei numeri, ma che non contengono date: in questo caso è necessario effettuare un'analisi sul viario e raccogliere tutti gli indirizzi che presentano tale anomalia. Quando ci si troverà di fronte ad una situazione del genere si può agire in questo modo: si eliminano dalle stringhe tutti i numeri, sopprimendo quindi anche il

numero civico, e si esegue la funzione di similarità tra l'indirizzo postale e l'insieme di toponimi che presentano l'anomalia descritta in precedenza. Se si trova un toponimo che ha una somiglianza molto alta, si contano in esso le occorrenze delle stringhe composte da uno spazio seguito da un numero, e poi si taglierà l'indirizzo di partenza in base al numero di queste occorrenze.

▪ 3.3 Cenni ad altri possibili algoritmi

Distanza di Levensthein

Questo algoritmo è stato inventato nel 1965 dallo scienziato sovietico Vladimir Levensthein, ed è anche chiamato edit distance (distanza di editing). Questa distanza rappresenta il numero minimo di modifiche elementari da apportare ad una stringa A per trasformarla in una stringa B.

Per modifica elementare si intende:

- La cancellazione di un carattere
- La sostituzione di un carattere con un altro
- L'inserimento di un carattere

L'algoritmo comune richiede l'uso di una matrice di $(n+1) \times (m+1)$, dove m e n sono le lunghezze delle due stringhe, quindi il tempo di calcolo è pari a $O(mn)$. Usando alcuni accorgimenti però è possibile vedere che si può implementare l'algoritmo usando una matrice di due sole righe, che contenga la precedente e la corrente. Il tempo di calcolo si riduce quindi a $O(n)$.

Per esempio, il confronto tra le stringhe "casa" e "cassa" restituisce il valore 1, in quanto c'è un solo carattere di differenza tra le due stringhe.

Soundex

Soundex è un algoritmo fonetico, pensato appositamente per la lingua inglese. Lo scopo di tale algoritmo è fornire una rappresentazione approssimativa di come un termine, una parola, viene pronunciata. E' stato pensato e provato soprattutto per i cognomi, in quanto è stato usato inizialmente per il primo censimento della popolazione degli U.S.A. nel 1880.

La procedura può essere utilizzata per semplificare le ricerche nei database, in cui si sa come viene pronunciato il testo, ma non si sa come viene scritto. Ad esempio i cognomi "Smith" e "Smyth" hanno la stessa pronuncia, quindi verranno identificati nello stesso modo. Questo algoritmo inoltre riduce il problema di abbinamento quando il testo contiene degli errori di battitura.

Il risultato è dato da un carattere ed un insieme di tre numeri. L'algoritmo può essere descritto secondo i seguenti passi:

- Il primo carattere rimane uguale
- Si scartano i caratteri A, E, I, O, U, Y, W e H, a meno che non siano iniziali
- Si traducono i restanti caratteri:
B, F, P, V → 1
C, G, J, K, Q, S, X, Z → 2
D, T → 3
L → 4
M, N → 5
R → 6
- Scartare le lettere adiacenti che hanno lo stesso codice

- Prendere i primi quattro caratteri, eliminando i restanti a destra. Nel caso il risultato sia più corto di quattro caratteri, si aggiungono degli zeri in fondo

Come si può vedere, le consonanti dal suono simile vengono tradotte con lo stesso numero.

Adottare questo algoritmo per la lingua italiana però richiederebbe uno studio molto accurato, eseguendo dei test su migliaia e migliaia di parole. Per prima cosa andrebbe impostata una diversa tabella di corrispondenze tra le consonanti e i numeri. Inoltre nella nostra lingua c'è il problema delle doppie, che in inglese possono essere facilmente trascurabili; studiare come le vocali influenzano la pronuncia delle consonanti adiacenti; il modo in cui influiscono gli accenti, e problematiche simili a queste.

4. IL SISTEMA SVILUPPATO

4.1 Il sistema GIS

Un GIS (*Geographic Information System*, Sistema Informativo Geografico), è un sistema informativo software che permette, in maniera del tutto automatica, l'acquisizione, l'analisi, la visualizzazione e la restituzione di informazioni derivanti da dati geografici, o dati spaziali. Per dato spaziale si intende un'informazione associata ad un sistema di coordinate georeferenziate, cioè relativo ad una specifica posizione sulla superficie terrestre. In termini più semplici, può essere pensato come una fusione tra:

- Cartografia, che può essere gestita mediante dei sistemi di disegno computerizzati, tipicamente CAD (Computer Aided Design)
- Analisi statistica
- Sistemi di database relazionali, ovvero i DBMS (DataBase Management System)

La versatilità offerta da questo innovativo sistema, rispetto ad altri sistemi informativi, fa del GIS uno strumento indispensabile per la pianificazione territoriale e per tutte le operazioni di visualizzazione ed analisi relative alla gestione del territorio nei suoi più disparati aspetti.

Può essere usato ad esempio in archeologia, nell'urbanistica, nello studio della demografia, nella programmazione della mobilità urbana, nel geomarketing (ovvero la pianificazione, secondo informazioni riferite al territorio, delle attività di marketing), e molti altri settori.



Figura 5 Elementi fondamentali di un GIS

Gli elementi fondamentali di un GIS sono:

- Hardware
- Software
- Dati
- Metodi
- Utenti

Caratteristica fondamentale di un GIS è la sua capacità di georeferenziare i dati, ovvero di attribuire ad ogni elemento le sue coordinate spaziali reali. In altre parole, le coordinate di un oggetto non sono memorizzate relativamente ad un sistema di riferimento arbitrario (ad esempio 12 centimetri dal bordo inferiore e 5 da

quello sinistro di una mappa) né relativamente al sistema di coordinate della periferica usata, come la tavoletta digitalizzatrice o il video, ma sono memorizzate secondo le coordinate del sistema di riferimento in cui è realmente situato l'oggetto, ovvero la sua misura in gradi di longitudine e di latitudine, e nelle reali dimensioni, non in scala. La scala di rappresentazione diventa a questo punto solamente un parametro per definire il grado di accuratezza e la risoluzione delle informazioni grafiche.

Sulle carte geografiche, che sono piane, tali posizioni sono riferite ad un Sistema di Coordinate Planari, ovvero alla loro distanza da un'origine misurata lungo due assi (x, y). Per "trasferire" gli elementi dalla superficie sferica della terra a quella piana delle carte, si utilizzano delle proiezioni cartografiche, ovvero un modello matematico in grado di trasformare latitudine e longitudine nel sistema di coordinate x e y. Le proiezioni più usate sono l'U.T.M. (Universal Transverse of Marcator), la Gauss-Boaga e la Lambert.

L'archiviazione dei dati, una volta definito il sistema di riferimento ed il modello dei dati, avviene normalmente utilizzando due formati: vettoriale e raster. Per dati vettoriali si intendono dati geometrici memorizzati attraverso le coordinate dei punti significativi degli elementi stessi: ad esempio un cerchio potrebbe essere memorizzato attraverso le coordinate del suo centro e la misura del suo raggio. Tipici dati memorizzati in questo formato sono quelli che provengono dalla digitalizzazione manuale di mappe, dai rilievi topografici, dai CAD e dai GPS (Global Positioning System).

Per dati raster si intendono invece dei dati memorizzati tramite la creazione di una griglia regolare in cui ad ogni cella (assimilabile ad un pixel) viene assegnato un valore alfanumerico che ne rappresenta un attributo. Questi valori possono rappresentare sia singoli fenomeni naturali od antropici (come temperatura od uso del suolo), sia il risultato della combinazione di più informazioni mediante metodologie di analisi, od anche semplicemente attributi grafici come la tonalità di grigio od il colore.

Tipici dati raster sono quelli generati dagli scanner e dai programmi di interpretazione di immagini, come quelli utilizzati per le immagini da satellite.

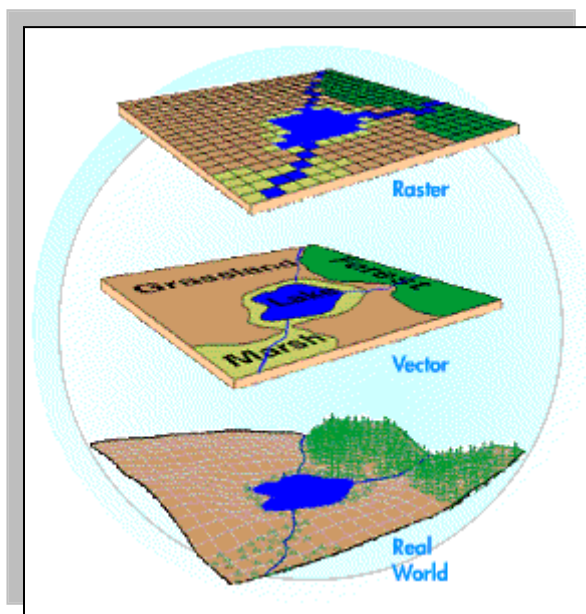


Figura 6 Integrazione di dati raster e vettoriali

Il formato raster e quello vettoriale costituiscono due differenti approcci alla rappresentazione dei dati geografici: la scelta è legata solo a problemi di funzionalità e disponibilità di cartografia. I principali software GIS consentono di strutturare in forma raster basi di dati vettoriali e viceversa così da sfruttare appieno le potenzialità delle due tipologie in base alle esigenze del progetto.

In ogni caso l'elemento più importante del modello dati di un GIS rimangono sempre gli attributi. Infatti un GIS ha il suo obiettivo principale nell'analisi dei dati, per diventare uno strumento di supporto alle decisioni. L'utente quindi non ha solo bisogno di restituire una carta delle zone edificate, quanto di rappresentare un tematismo, ad esempio, retinarla in funzione dell'età media della

popolazione residente. Per ottenere questo, si potrebbe interrogare una banca dati di tipo relazionale: per esempio, ad ogni edificio potrebbe essere associato un indirizzo, mentre in un altro data base (quello anagrafico) ad ogni indirizzo potrebbe essere associata la data di nascita delle persone che vi risiedono. Tramite quindi l'informazione "Indirizzo" si potrebbe creare una relazione tra i due data base ed effettuare un'analisi relativa all'età media restituendola poi graficamente su carta. Dalla stessa analisi si potrebbe poi derivare un nuovo tematismo relativo all'anzianità dei residenti e utilizzandolo in sovrapposizione al tematismo relativo alla carta dei bacini d'utenza dei centri anziani del Comune o dei servizi materno-infantili dell'Unità

Sanitaria Locale, verificare la congruenza tra bacini d'utenza dei servizi (offerta) e potenziale utenza (domanda) per poi decidere eventualmente un nuovo piano d'azione o di sviluppo. Gli attributi che possono risiedere anche su più sistemi ed essere aggiornati da molti applicativi (nel caso precedente sono aggiornati dall'ufficio anagrafe e utilizzati da quello di pianificazione

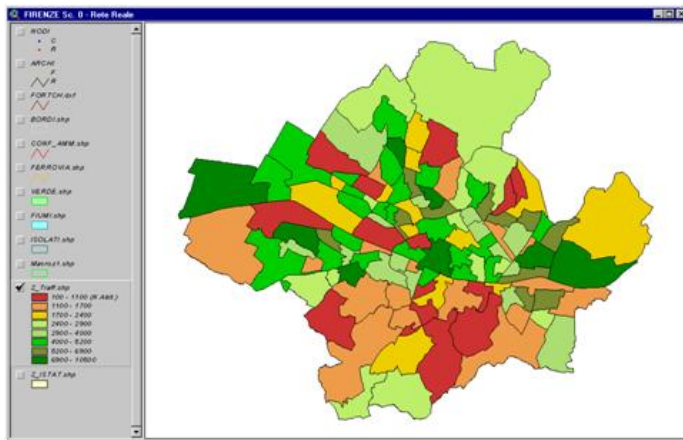


Figura 7 Un esempio di tematismo, rappresentante la densità di un territorio

dei servizi), sono in genere memorizzati su dei data base relazionali ed interrogabili mediante linguaggi di tipo SQL (Structured Query Language).

Il GIS consente di mettere in relazione tra loro dati diversi, sulla base del loro comune riferimento geografico in modo da creare nuove informazioni a partire dai dati esistenti. Esso offre ampie possibilità di interazione con l'utente e un insieme di strumenti che ne facilitano la personalizzazione e l'adattamento alle problematiche specifiche dell'utente.

Normalmente presentano delle funzionalità di analisi spaziale, ovvero di trasformazione ed elaborazione degli elementi geografici degli attributi. Esempi di queste elaborazioni sono:

- L'overlay topologico: in cui si effettua una sovrapposizione tra gli elementi dei due temi per creare un nuovo tematismo (ad esempio per sovrapporre il tema dei confini di un parco con i confini dei comuni per determinare le superfici di competenza di ogni amministrazione o la percentuale di area comunale protetta).
- Le query spaziali, ovvero delle interrogazioni di basi di dati a partire da criteri spaziali (vicinanza, inclusione, sovrapposizione etc.).
- Il buffering: da un tema puntuale, lineare o poligonale definire un poligono di rispetto ad una distanza fissa o variabile in funzione degli attributi dell'elemento.
- La segmentazione: algoritmi di solito applicati su temi lineari per determinare un punto ad una determinata lunghezza dall'inizio del tema.
- La network analysis: algoritmi che da una rete di elementi lineari (es. rete stradale) determinano i percorsi minimi tra due punti.

▪ 4.2 La struttura generale

Il sistema è costituito da tre strumenti software diversi:

- Ambiente GIS (ESRI ArcView3.1): è l'ambiente che costituisce il componente fondamentale riguardo all'interfaccia utente: è tramite esso, infatti, che vengono attivate le procedure che permettono all'operatore di interagire con il sistema per introdurre e/o modificare dati, visualizzare ed analizzare i risultati ottenuti;
- Database (MS-Access): con esso vengono memorizzati i dati caratteristici della rete oggetto di studio, tali dati vengono poi passati ai modelli di simulazione.
- MS-VisualBasic 6.0 ed ODBC: buona parte dell'interfaccia utente è sviluppata in ambiente Visual Basic che ha fatto da tramite tra il database e l'ambiente GIS. In particolare, Visual Basic è stato impiegato laddove era richiesta la realizzazione di procedure di calcolo e di dialogo non banali, i dialog di ArcView sono invece stati utilizzati per risolvere situazioni più semplici senza appesantire il sistema. Trasversale ai

due ambienti è stato poi l'utilizzo di opportune ODBC per l'accesso e la manipolazione del database Access.

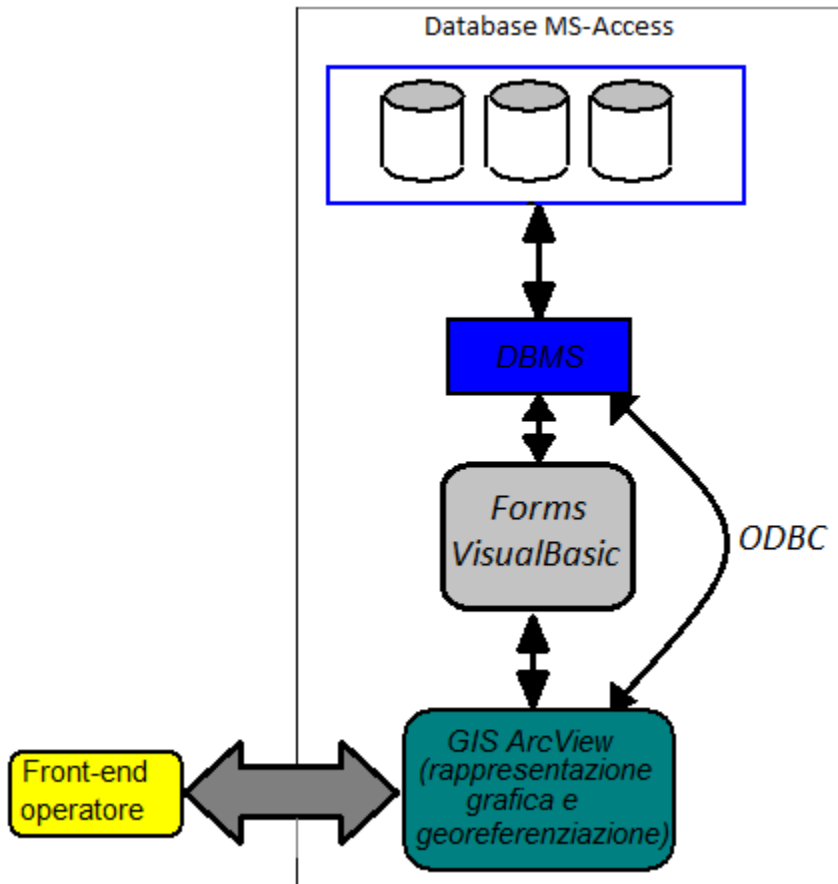


Figura 8 Struttura generale del sistema

Per quanto riguarda la struttura delle directory che il sistema utilizza, si può fare riferimento alla figura 9, dove:

- **C: >.** Rappresenta la root del disco su cui è installato il sistema (per comodità si è fatto riferimento all'unità C).
- **LinDes.** È la radice del sistema di directory del progetto, essa contiene le seguenti sottodirectory:
 - **Base.** Contiene tutti i files di base GIS relativi alla città cartografica sullo sfondo e alle zonizzazioni.
 - **Data.** Contiene il database alfanumerico della città di Livorno.
 - **Exe.** Contiene tutti gli eseguibili del progetto (forms di interfacciamento tra i vari ambienti) sviluppati in ambiente VisualBasic.
 - **Rete.** Contiene i file GIS relativi alla rete stradale (Nodi, Archi, Links). Nello specifico essa contiene tante sottodirectory quanti sono gli scenari gestiti. La sottodirectory Scen0 è riservata alla rete reale ed è l'unica che deve essere sempre presente.
 - **Work.** È la directory di lavoro del modulo GIS, essa contiene i file temporanei gestiti dal sistema e periodicamente viene cancellata in modo automatico.

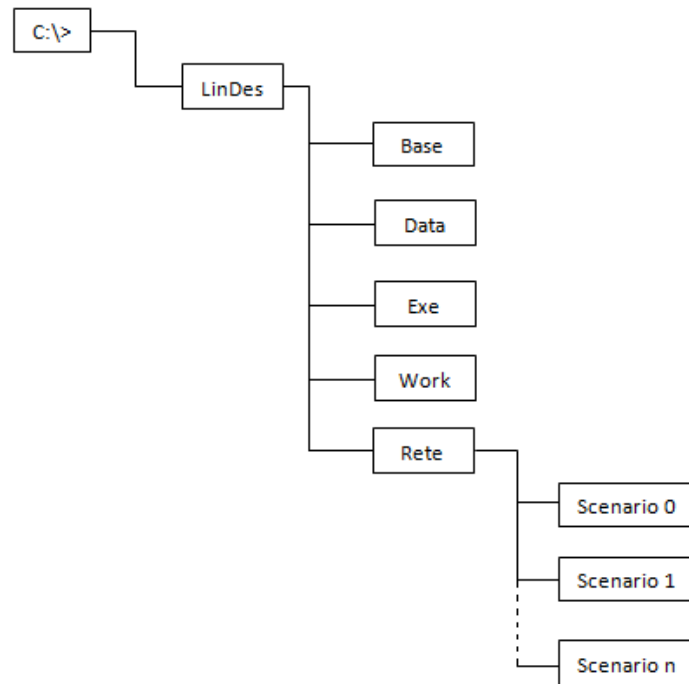


Figura 9 La struttura delle directory

4.3 Il database alfanumerico

Come già accennato in precedenza, il sistema di DBMS usato per la gestione del database alfanumerico è MS-Access.

La struttura centrale del database è relativa ai dati di una città, in particolar modo la città di Livorno.

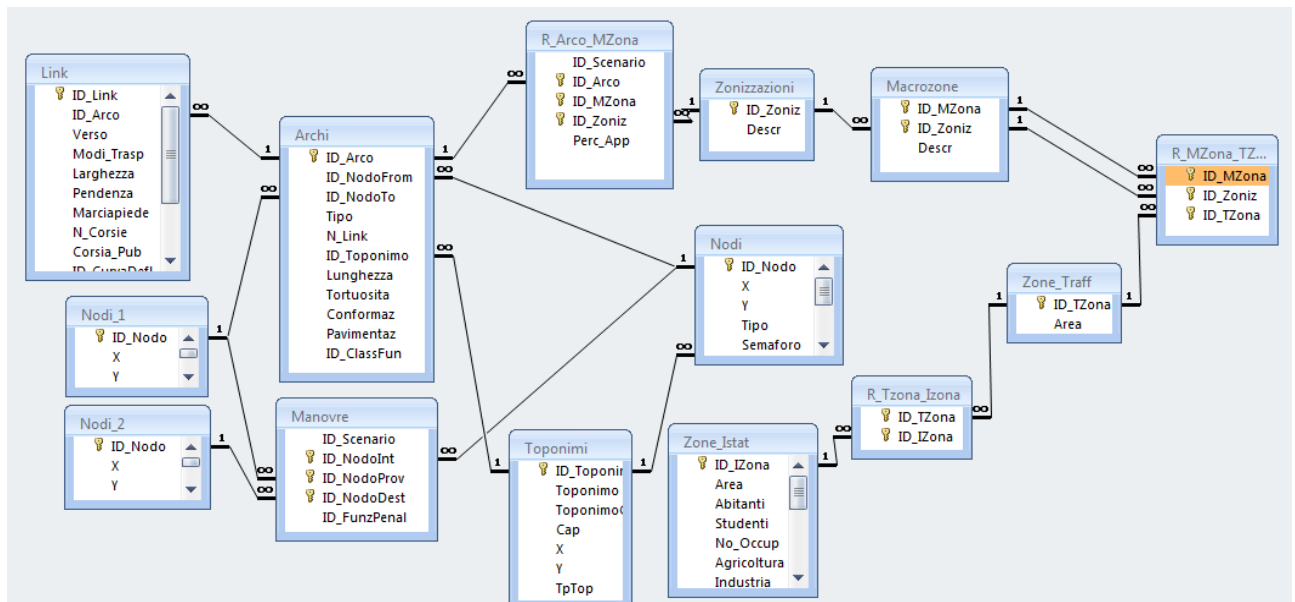


Figura 10 Schema delle relazioni interne al database

Le principali tabelle sono quindi:

- **Nodi.** Contiene l'insieme di tutti i nodi, descritti da due coordinate (x, y). La sua chiave primaria è l'attributo ID_Nodo.
- **Archi.** Contiene l'insieme di tutti gli archi, ed è in doppia associazione con la tabella dei nodi, in quanto deve memorizzare il nodo da cui parte (ID_NodoFrom) e il nodo in cui arriva (ID_NodoTo). La sua chiave primaria è l'attributo ID_Arco.

- **Manovre.** Contiene l'insieme delle manovre che si possono effettuare ad un incrocio, ed è in tripla associazione con la tabella dei nodi, in quanto deve memorizzare il nodo di provenienza (ID_NodoProv), il nodo destinazione (ID_NodoDest) ed il nodo sul quale si effettua la manovra (ID_NodoInt). La chiave primaria è formata dall'insieme di questi tre attributi.
- **Link.** Contiene tutti i sensi di marcia di un arco. L'associazione che ha con la tabella Archi è 1 a 1, nel caso il senso di percorrenza sia uno solo, oppure 1 e 2, nel caso i sensi di percorrenza siano due. In questa tabella inoltre sono contenuti tutti gli attributi della strada, ovvero la larghezza, la pendenza, i tipi di mezzo di trasporto che possono percorrerla, il numero di corsie, ecc. La chiave primaria è l'attributo ID_Link.
- **Toponimi.** Contiene tutti i toponimi della città, la chiave primaria è l'attributo ID_Toponimo.
- **Zone_Istat.** Contiene tutte le informazioni riguardanti le zone Istat della città, ovvero l'area, in numero di abitanti, il numero di studenti, ecc. La chiave primaria è ID_Zona.
- **Zone_Traff.** Contiene l'elenco di tutte le zone di traffico, la chiave primaria è l'attributo ID_TZona.
- **R_Tzona_Izona.** È la tabella che viene fuori dall'associazione m:n che persiste tra le tabelle Zone_Istat e Zone_Traff, ed indica quali zone Istat fanno parte di ogni zona di traffico. La chiave primaria è formata dall'insieme delle due chiavi esterne, ID_TZona e ID_Izona.

Tutte le altre tabelle presenti serviranno per sviluppi futuri del sistema.

È necessario tenere separato questo database alfanumerico dal database prettamente geografico, in quanto le informazioni qui contenute potrebbero servire anche ad altre applicazioni, e non esclusivamente al sistema GIS. È però opportuno tenere sempre allineati i dati territoriali ed i dati alfanumerici, in modo che non si creino incongruenze. Per mantenere questa integrità, si devono usare degli strumenti per il controllo, come gli ODBC ed il linguaggio VisualBasic.

▪ 4.4 Utilizzo di ODBC e VisualBasic come integratori di ambienti

ODBC

Con il termine ODBC (Open DataBase Connectivity) si intende un programma di interfaccia che permette alle applicazioni di accedere ai dati gestiti da un DBMS (Data Base Management System) che usi l'SQL (Structured Query Language) come linguaggio standard per la gestione dei dati. Per far funzionare il sistema quindi, c'è il bisogno di definire un ODBC MS-Access (tramite il Pannello di controllo di Windows) per fare accedere il GIS al database alfanumerico. Nel linguaggio Avenue (il linguaggio di scripting integrato in ArcView 3.1) è possibile connettersi all'ODBC usando il seguente frammento di codice:

```
_connODBC=SQLCon.Find(_NomeCitta)
if (_connODBC=Nil) then
    msgbox.Info("Necessario definire ODBC per "+_NomeCitta,"ODBC")
    av.quit
else
    _connODBC.Login("user/password")
end
```

Una volta effettuata la connessione al database, si può intervenire direttamente su di esso tramite delle istruzioni SQL. Ad esempio si possono fare degli inserimenti o delle cancellazioni, usando il metodo `ExecuteSQL()`, passando come parametro una stringa contenente l'istruzione.

Due esempi possono essere i seguenti:

- Cancellazione

```
AString="DELETE * FROM R_Arco_MZona WHERE Id_Zonizz = 1"  
_ConnODBC.ExecuteSQL(AString)
```
- Inserimento

```
AString="INSERT INTO R_Arco_MZona (Id_arco, Id_MZona,  
Id_Zoniz, Perc_App) VALUES (" + Ida.asstring + ",  
" + Idn.asstring + ", " + Idz.asstring + ", " + Ipp.asstring + ") "  
_ConnODBC.ExecuteSQL(AString)
```

VisualBasic

Il VisualBasic è stato utilizzato per situazioni più complesse, dove si accede al database tramite delle forms di dialogo, che permettono all'utente di interagire in modo puntuale con le singole tabelle per operazioni di data-entry dettagliate.

L'utilizzo di una form VisualBasic in uno script Avenue comporta due passi fondamentali:

- Lancio della form VisualBasic
- Passaggio dei dati tra il GIS e la form VisualBasic

L'esecuzione di una form dall'ambiente Avenue prevede due diversi tipi di modalità: sincrona ed asincrona.

Nel primo caso, il processo chiamante (Avenue) si blocca e passa il controllo al processo chiamato (form VisualBasic), che solo al termine della propria esecuzione cede di nuovo il controllo al chiamante. Questo tipo di esecuzione non prevede un passaggio di dati tra i due processi. Per eseguire un processo con questa modalità si usa l'istruzione: `System.ExecuteSynchronous(_pathexe+<nome Processo>)`, dove `_pathexe` è una variabile globale che indica il percorso della directory che contiene i file eseguibili.

Nel caso di esecuzione asincrona, invece i due processi (il chiamato -server- ed il chiamante -client-) rimangono attivi indipendentemente l'uno dall'altro con possibilità di dialogo dinamico realizzato tramite il meccanismo DDE (Data Dynamic Exchange). Con questo tipo di esecuzione, è necessario controllare costantemente via software la congruenza dei contesti operativi dei due processi perché, data la loro mutua interazione, può succedere che ad esempio uno dei due processi richieda all'altro l'esecuzione di una operazione valida in un contesto che non è più quello corrente perché nel frattempo modificato dall'utente o dal qualche altro evento software che si sia venuto a determinare.

Il codice per utilizzare questo tipo di conversazione è il seguente:

```
aDDE=DDEClient.make("CancArch","Transfer")  
if (aDDE.haserror) then  
    System.Execute(_pathexe+"CancArch.exe")  
    aDDE=DDEClient.make("CancArch","Transfer")  
    if (aDDE.haserror) then  
        msgbox.info("Errore in DDE CancArch","")  
        exit  
    end  
end  
aDDE.execute(Param)
```


L'istruzione `aDDE=DDEClient.make("CancArch","Transfer")` serve ad attivare preventivamente la connessione tra i due processi, che verrà attivata in modo vero e proprio con l'istruzione `System.Execute(_pathexe+"CancArch.exe")`. Dalla parte VisualBasic il parametro fornito dal Gis viene intercettato dalla subroutine "Form_LinkExecute".

5. I DATI TERRITORIALI

5.1 La banca dati geografica

Un dato viene definito geografico quando può essere associato ad un luogo sulla superficie terrestre mediante l'attribuzione di una coppia di coordinate (latitudine, longitudine). La banca dati geografica è costituita da un insieme di dati geografici e non, fra loro correlati, riferiti alla medesima area geografica e organizzati per uno scopo specifico.

In ArcGis 3.1, i dati territoriali sono contenuti in appositi files chiamati *Shapefile*. Con Shapefile si intende un insieme di file con estensione .shp, .dbf, .shx, ed altre, che hanno tutte in comune il prefisso dei nomi.

Spesso con Shapefile si intende soltanto il file con formato .shp, tuttavia questo file è incompleto, poiché interpretazione ed utilizzo dipendono dagli altri file. Questi file descrivono dati di tipo vettoriale "primitivi", che possono essere di tre tipi:

- Una coppia di coordinate (x, y) (**Punti**)
- Una sequenza di punti (**Linea**)
- Una sequenza di linee chiuse che non si incrociano (**Poligoni o Aree**)

Da sole queste primitive, dette "Feature", non sono utili, in quanto mancano degli attributi che specificano cosa queste primitive rappresentino.

Perciò una tavola di records registra proprietà e attributi per ogni primitiva "Shape" dello Shapefile. Le Shape, insieme ai dati attributari, possono creare infinite rappresentazioni di dati geografici, da cui a loro volta dipende la potenza e l'accuratezza delle analisi geospaziali che sono il fine dei GIS.

Come detto prima, uno Shapefile è costituito da un insieme di file; ognuno di essi condivide con gli altri il proprio nome (prefisso prima del punto) e sono collocati tutti nella stessa cartella. Tre di questi file sono obbligatori:

- **.shp**, conserva le geometrie;
- **.shx**, conserva l'indice delle geometrie;
- **.dbf**, il database degli attributi.

Gli Shapefile non possono contenere informazioni topologiche (adiacenza, connessione, prossimità, coincidenza) sulle loro feature, essi sono quindi la mera rappresentazione spaziale delle feature e degli attributi.

Il file .dbf non è in grado di gestire i valori NULL dei campi, quindi gli attributi non saranno contenuti tutti in questo file, ma si farà riferimento al database alfanumerico descritto precedentemente, effettuando delle join opportune per compattare un insieme di più tabelle in una sola.

5.2 La zonizzazione: Zone Istat, Zone di traffico e Macrozone

Attraverso la zonizzazione è possibile raggruppare tutti i luoghi di origine e di destinazione distribuiti sul territorio comunale in un numero di Bacini Origine/Destinazione (BOD) coincidenti con le 347 Zone ISTAT del Comune di Livorno. All'interno dei BOD è individuato un punto, il centroide (baricentrico rispetto alla distribuzione interna della popolazione e delle attività, rappresentato come un punto di coordinate x, y), al quale sono riportati tutti gli spostamenti che hanno origine o destinazione nello stesso BOD. All'interno di ogni BOD è

descritta una rete stradale sulla quale possono essere schematizzati gli spostamenti che hanno origine e destinazione in quella zona.

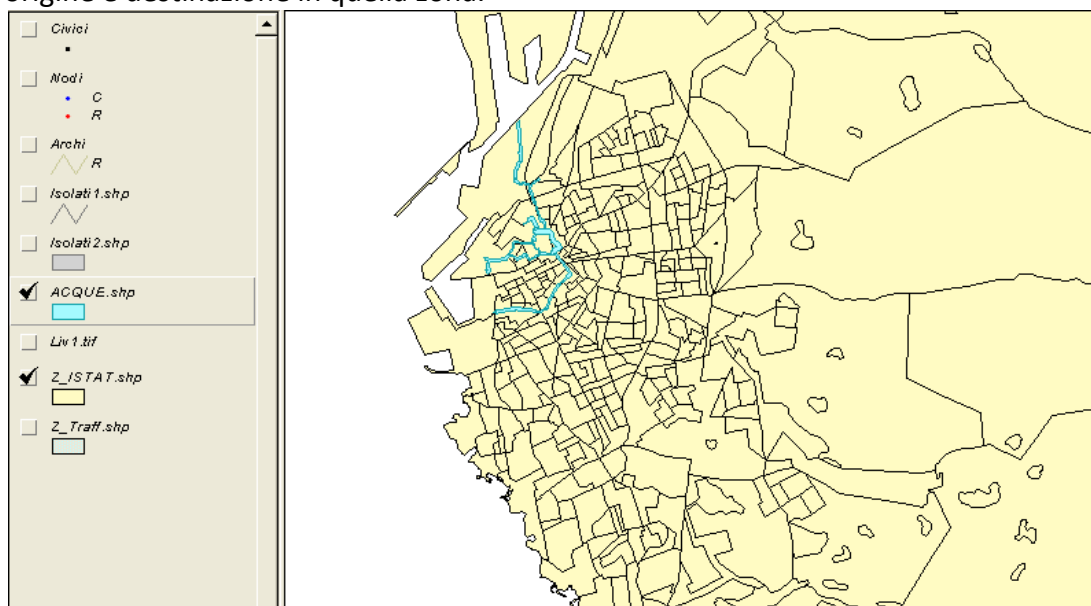


Figura 11 Le zone Istat di una parte di Livorno

La zonizzazione costituisce la base tecnica necessaria per la descrizione, la quantificazione, la distribuzione spaziale della domanda di mobilità.

E' evidente però che l'area destinata allo studio degli spostamenti non può essere circoscritta solamente al comune di Livorno, ma deve essere estesa ad aree al di fuori di esso, individuando dei centroidi rappresentativi per le origini degli spostamenti esterni all'area di studio. Questi centroidi esterni prendono il nome di "macrozone", in quanto raggruppano molte località di provenienza.

Nello studio effettuato, abbiamo preso in considerazione tre diverse macrozone, individuate a seconda della posizione geografica rispetto al comune:

- A Nord è individuata una zona dalla quale provengono studenti dal comune di Pisa, che comprende anche le frazioni di Marina di Pisa e Tirrenia, e dalla provincia di Lucca.
- Ad Est è individuata una zona dalla quale provengono studenti sia dalla provincia di Livorno, come dal comune di Collesalveti, che dalla provincia di Pisa, come dai comuni della Valdera (Ponsacco, Pontedera, Santa Maria a Monte, ecc).
- A Sud è individuata una zona dalla quale provengono studenti soprattutto dalla provincia di Livorno, come dai comuni di Rosignano, Cecina e Piombino, ma anche dalla provincia di Grosseto, come dal comune di Follonica.

Ad Ovest non si individuano macrozone, in quanto il comune di Livorno è confinante con il Mar Tirreno (secondo i confini dell'*Organizzazione idrografica internazionale*, mentre per i confini dell'*Istituto Idrografico della Marina Militare Italiana* il comune sarebbe confinante con il Mar Ligure).

Le macrozone potevano anche essere suddivise in più sezioni, prendendo come riferimenti, oltre a quelli già descritti, anche la zona Nord-Est e la zona Sud-Est, in modo da partizionare in modo preciso il territorio. Questo però è soltanto un piccolo particolare, che dipende dallo studio che si vuole fare, ma che comunque è sempre possibile da realizzare senza nessuna difficoltà.

Per rendere poi più significative le informazioni portate dalle linee di desiderio, si aggredheranno più zone ISTAT, in quanto esse sono molto limitate nelle dimensioni a tal punto

da essere poco significative (dal punto di vista del numero di persone), in modo da formare delle zone di traffico.

Le zone ISTAT vengono quindi raggruppate per formare una zona che comprenda un numero maggiore di persone, e che quindi produca un maggior numero di spostamenti; nell'aggregare le zone, si deve seguire un certo criterio: ad esempio si possono raggruppare secondo i quartieri della città; oppure, in modo ancora migliore, si possono raggruppare secondo le linee dei pullman urbani, costituendo quindi le "zone di traffico": per fare questa operazione al meglio c'è bisogno comunque di uno specialista, ad esempio un trasportista, che conosca in modo notevole il territorio studiato. Per ogni zona di traffico è comunque definito il centroide, sempre sotto forma di coordinate (x,y), in modo da dare un punto indicativo alla partenza degli spostamenti.

Macrozona Nord

Comune	Provincia
Calci	Pisa
Cascina	Pisa
Pisa	Pisa
San Giuliano Terme	Pisa
Vecchiano	Pisa
Lucca	Lucca
Massarosa	Lucca
Viareggio	Lucca

Macrozona Est

Comune	Provincia
Bientina	Pisa
Buti	Pisa
Calcinaia	Pisa
Capannoli	Pisa
Casciana Terme	Pisa
Castelfranco	Pisa
Castellina Marittima	Pisa
Chianni	Pisa
Crespina	Pisa
Fauglia	Pisa
Lari	Pisa
Lorenzana	Pisa
Montecatini Valdicecina	Pisa
Montescudaio	Pisa
Orciano Pisano	Pisa
Peccioli	Pisa
Pomarance	Pisa
Ponsacco	Pisa
Pontedera	Pisa
Riparbella	Pisa
Santa Luce	Pisa
Santa Maria a Monte	Pisa
Terricciola	Pisa
Volterra	Pisa
Collesalveti	Livorno

Macrozona Sud

Comune	Provincia
Bibbona	Livorno
Campiglia Marittima	Livorno
Castagneto Carducci	Livorno
Cecina	Livorno
Piombino	Livorno
Rosignano Marittimo	Livorno
San Vincenzo	Livorno
Sassetta	Livorno
Surveto	Livorno
Follonica	Grosseto
Gavorrano	Grosseto
Massa Marittima	Grosseto
Monterotondo Marittimo	Grosseto

▪ 5.3 Trasferimento dei dati relativi alle scuole sul sistema GIS

Per collocare correttamente sulla mappa la residenza di ogni singolo alunno, si devono effettuare prima alcuni passi fondamentali.

La prima problematica riguarda i numeri civici: la tabella che li contiene, oltre al numero e le coordinate nel quale si trova fisicamente, non comprende delle informazioni fondamentali, come ad esempio il toponimo. Per prima cosa quindi c'è bisogno di eseguire una query spaziale, proprio tramite il GIS, che associ ad ogni civico un arco (ID_Arco), e di conseguenza un toponimo. Dopodiché si esegue un'altra query spaziale, in modo da associare ad ogni civico, oltre al toponimo, anche la zona Istat nella quale si trova. A questo punto si può fare un matching, anche qui tramite una query, tra gli indirizzi di residenza degli studenti ed i numeri civici. Gli studenti che invece non facevano parte del comune di Livorno, sono stati assegnati alla rispettiva macrozona di provenienza, in base alle tabelle presentate nella sezione precedente.

Alla fine di queste operazioni, si può riscontrare che 225 record non hanno trovato la corrispondenza con il numero civico: questa situazione si è venuta a creare perché i numeri civici che si posseggono non sono tutti quelli esistenti nel comune, ma sono soltanto una parte. Per migliorare le cose, si potrebbe pensare di scrivere uno script che assegni una collocazione approssimativa a questi record, cercando se ci sono dei numeri civici che sono molto vicini a quello che stiamo cercando.

Comunque sia, 225 record corrispondono a circa il 15,8% del totale, ma considerando i record che non sono stati normalizzati (ovvero che non hanno un toponimo corretto), i record che non hanno collocazione sulla mappa sono 180, ovvero circa il 12,7%, che è sempre una quantità abbastanza considerevole.

Un modello matematico che riesca a collocare approssimativamente questi record è comunque sviluppabile ed integrabile nel sistema, in modo da ottimizzare i risultati finali che si ottengono.

6. LA MOBILITA' NEL COMUNE DI LIVORNO

▪ 6.1 La creazione della matrice O/D e delle linee di desiderio

Per creare le linee di desiderio, si deve prima costruire la matrice delle Origini/Destinazioni (O/D), nella quale, l'origine è una zona di traffico e la destinazione è l'ITIS Galileo Galilei, a cui è stato assegnato il nodo numero 800.

Per la creazione di questa matrice si devono eseguire delle query SQL sul database alfanumerico. La prima cosa da fare è contare quanti studenti ci sono in ogni singola zona Istat; la query risultante sarà la seguente:

```
SELECT ID_I Zona, Count(*) AS Studenti INTO NStudi
FROM ITIS
GROUP BY ID_I Zona
```

Grazie a questa adesso, possiamo creare la matrice O/D in base alle zone di traffico. Per ottenerla, la query risultante dovrà fare una join tra la tabella ottenuta precedentemente e la tabella che contiene le associazioni tra zone Istat e zone di traffico, per poi sommare il numero di studenti proveniente da ogni zona, stabilendo così in numero degli spostamenti.

Il risultato sarà quindi il seguente:

```
SELECT ID_TZona, SUM(Studenti) AS Spostamenti INTO NStudZT
FROM NStudi, R_Tzona_Izona
WHERE NStudi.ID_Izona=R_Tzona_Izona.ID_Izona
GROUP BY ID_Tzona
```

La tabella O/D che si ricava è proposta di seguito:

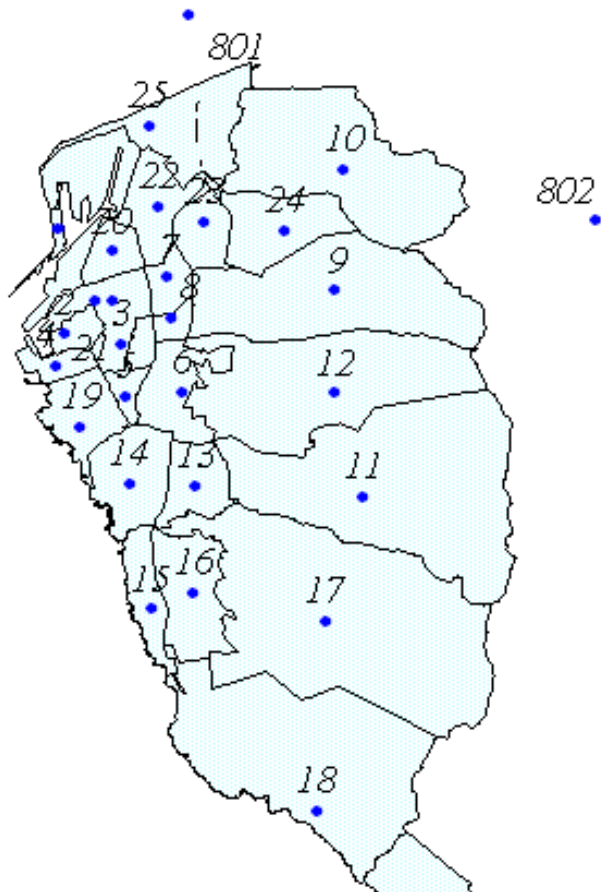


Figura 11 I nodi di partenza. A sud non è visibile il nodo 803, che rappresenta proprio la macrozona sud

Id_NodoFrom	Id_NodoTo	Spostamenti
2	800	50
3	800	116
4	800	40
5	800	36
6	800	83
7	800	97
8	800	53
9	800	6
11	800	18
12	800	21
13	800	55
14	800	51
15	800	31
16	800	10
17	800	12
19	800	81
20	800	24
21	800	130
22	800	19
23	800	13
801	800	15
802	800	165
803	800	63
Totale		1189

Nella Figura 11, è possibile vedere i nodi di partenza, in modo da avere un’idea di dove possono essere i nodi riportati nella matrice O/D.

Dopo la creazione della matrice, si deve passare alla creazione del tematismo delle linee di desiderio. Questa parte va realizzata creando un apposito script che sia in grado di disegnare le linee in base alla matrice O/D costruita.
Lo script che esegue queste operazioni è proposto nella pagina seguente.

```

'
'
' Script : Mob.LinDes'
' Funzione : Crea Tema Linee di desiderio
'
' Data ultima revisione: 03/06/11
'
'
'Elimina il tema dalla view se gia' esiste

ZTraff=_ViewLinDes.findtheme("Z_Traff.shp")
ZTraff.setvisible(True)

LinDes=_ViewLinDes.findtheme("LinDes.shp")
if (LinDes <> nil) then
    LinDes.setactive(false)
    LinDes.setvisible(false)
    _ViewLinDes.deletetheme(LinDes)
end
'--- Crea la FTAB LinDes e prende i fields
    TabLD = FTab.MakeNew((_pathrete+"LinDes.dbf").AsFileName,PolyLine)
    campoidLD = Field.Make("Id_LinDes",#FIELD_DECIMAL,5,0)
    campoNFrom = Field.Make("Id_NodoFrom",#FIELD_DECIMAL,5,0)
    campoNTo = Field.Make("Id_NodoTo",#FIELD_DECIMAL,5,0)
    campoSpost= Field.Make("Spostam",#FIELD_DECIMAL,6,0)

    TabLD.seteditable(true)
    TabLD.AddFields({campoidLD,campoNFrom,campoNTo,campoSpost})
    ThLD = FTheme.Make(TabLD)
    TbFLD =ThLD.getFtab
    campoShpLD=TbFLD.Findfield("Shape")
    campoidLD=TbFLD.Findfield("Id_LinDes")
    campoNFrom=TbFLD.Findfield("Id_NodoFrom")
    campoNTo=TbFLD.Findfield("Id_NodoTo")
    campoSpost=TbFLD.Findfield("Spostam")

    Query="SELECT * FROM Nodi WHERE Id_Nodo=800"
    theVTab=VTab.MakeSQL(_connODBC, Query)
    if (theVTab.GetNumRecords=0) Then
        msgbox.warning("Non Esistono Dati relativi alla Destinazione","Linee
        Desiderio")
        return nil
    end
    for each rec in theVTab
        xd=theVTab.returnvalue(theVTab.findField("x"),rec)
        yd=theVTab.returnvalue(theVTab.findField("y"),rec)
    end

    pd=point.make(xd,yd)
'-----
    Query="SELECT * FROM NStudT,Nodi WHERE Id_Tzona=Id_Nodo AND
    Id_Tzona>0"
    theVTab=VTab.MakeSQL(_connODBC, Query)
    if (theVTab.GetNumRecords=0) Then

```

```

    msgbox.warning("Non Esistono Dati ", "Linee Desiderio")
    return nil
end
' *-----
' Per ogni Linea di Desiderio
' *-----
nLD=0
for each rec in theVTab
    xo=theVTab.returnvalue(theVTab.findField("x"),rec)
    yo=theVTab.returnvalue(theVTab.findField("y"),rec)
    spost=theVTab.returnvalue(theVTab.findField("Studenti"),rec)
    nFrom=theVTab.returnvalue(theVTab.findField("Id_Tzona"),rec)
    po=point.make(xo,yo)
    linD=PolyLine.make(List.Make.Add({po,pd}))
    nLD=nLD+1
    'Si crea il record LD e lo si avvalora
    rLD=TbFLD.addRecord
    TbFLD.setValue(campoShpLD,rLD,linD)
    TbFLD.setValue(campoidLD,rLD,nLD)
    TbFLD.setValue(campoNFrom,rLD,nFrom)
    TbFLD.setValue(campoNTo,rLD,800)
    TbFLD.setValue(campoSpost,rLD,spost)
end
TbFLD.seteditable(false)
' creazione legenda
_ViewLinDes.AddTheme(ThLD)
ThLD.SetActive(true)
ThLD.SetVisible(true)
ThLD.invalidate(True)
Arr=1
Crmp=14
LegendaFlusso = ThLD.GetLegend
LegendaFlusso.SetLegendType(#LEGEND_TYPE_SYMBOL)
LegendaFlusso.Natural(ThLD,"Spostam",5)
'LegendaFlusso.DisplayNoDataClass(True)
Listasimboli = LegendaFlusso.getsymbols
theColorRamp=SymbolList.GetPreDefined(#SYMLIST_TYPE_COLORRAMP).Get(Crmp)
ListaSimboli.RampSavedColors(theColorRamp)
ListaSimboli.RampSizes(2,5)
LblClas=LegendaFlusso.getclassifications.get(0)
LegendaFlusso.setprecision(Arr)
WLbl=LblClas.GetLabel
LblClas.SetLabel(WLbl++"Studenti")
ListaSimboli = ThLD.getlegend.getSymbols
Nclas=(ListaSimboli.count)-1
'ListaSimboli.Get(Nclas).Setsize(0.5)
'ListaSimboli.Get(Nclas).Setcolor(_noval)
ThLD.setlegend(LegendaFlusso)
ThLD.updatelegend

```

L'algoritmo parte controllando se il tema delle linee di desiderio esiste già: se esiste viene cancellato, poi viene creato il nuovo tema, definendo una nuova FTab, contenente delle linee, che ha gli attributi:

- **ID_LinDes.** L'identificativo della linea di desiderio.
- **ID_NodoFrom.** L'identificativo del nodo di partenza.
- **ID_NodoTo.** L'identificativo del nodo di arrivo, che è sempre 800, il nodo dell'ITIS Galileo Galilei.
- **Spostam.** Numero di spostamenti per la linea.

Creata questa tabella, si passa al recupero delle coordinate spaziali del nodo 800 dal database alfanumerico, effettuando una query via ODBC, e si colloca sulla mappa. Il passo successivo è recuperare la tabella degli spostamenti, sempre dal database alfanumerico, effettuando la query `SELECT * FROM NStudT,Nodi WHERE Id_Tzona=Id_Nodo AND Id_Tzona>0`. Da questa si nota in particolare che l'identificativo della zona di traffico corrisponde all'identificativo del nodo di partenza della linea.

A questo punto si esegue un ciclo *for each*, in modo da scorrere tutti i record della tabella presa in precedenza, che rappresenta la matrice degli spostamenti. Per ogni occorrenza quindi, si disegna il nodo di partenza, le cui coordinate sono contenute proprio nella tabella che stiamo considerando, dopodiché si disegna la linea che va dal punto di partenza al punto di arrivo (800). Dopo aver disegnato la linea, si salvano tutti i dati nella tabella definita inizialmente, assegnando ad ogni linea creata un identificativo numerico crescente.

Lo script si conclude con la creazione della legenda. Il risultato ottenuto è visibile in figura 12.

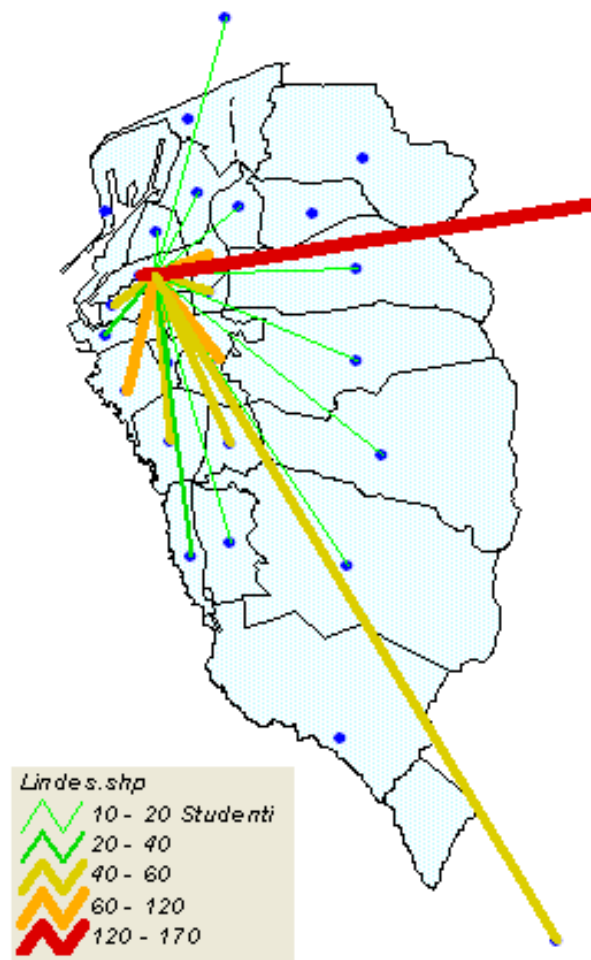


Figura 12 Le linee di desiderio

Il sistema inoltre fornisce un'utilità per il calcolo del cammino minimo da un nodo di partenza ad un nodo di destinazione, fornendo la sequenza degli archi da seguire, così da poter determinare l'effettivo percorso tra il punto di origine di una linea di desiderio ed il punto di arrivo.

La procedura usata per questo calcolo è l'algoritmo di Dijkstra, prendendo in considerazione, come costo di percorrenza, soltanto la lunghezza del cammino ed il mezzo utilizzato (mezzo pubblico, mezzo privato, oppure a piedi), senza valutare il tempo di percorrenza, che può essere determinato da fattori come la velocità massima consentita, la presenza o meno di semafori ecc.

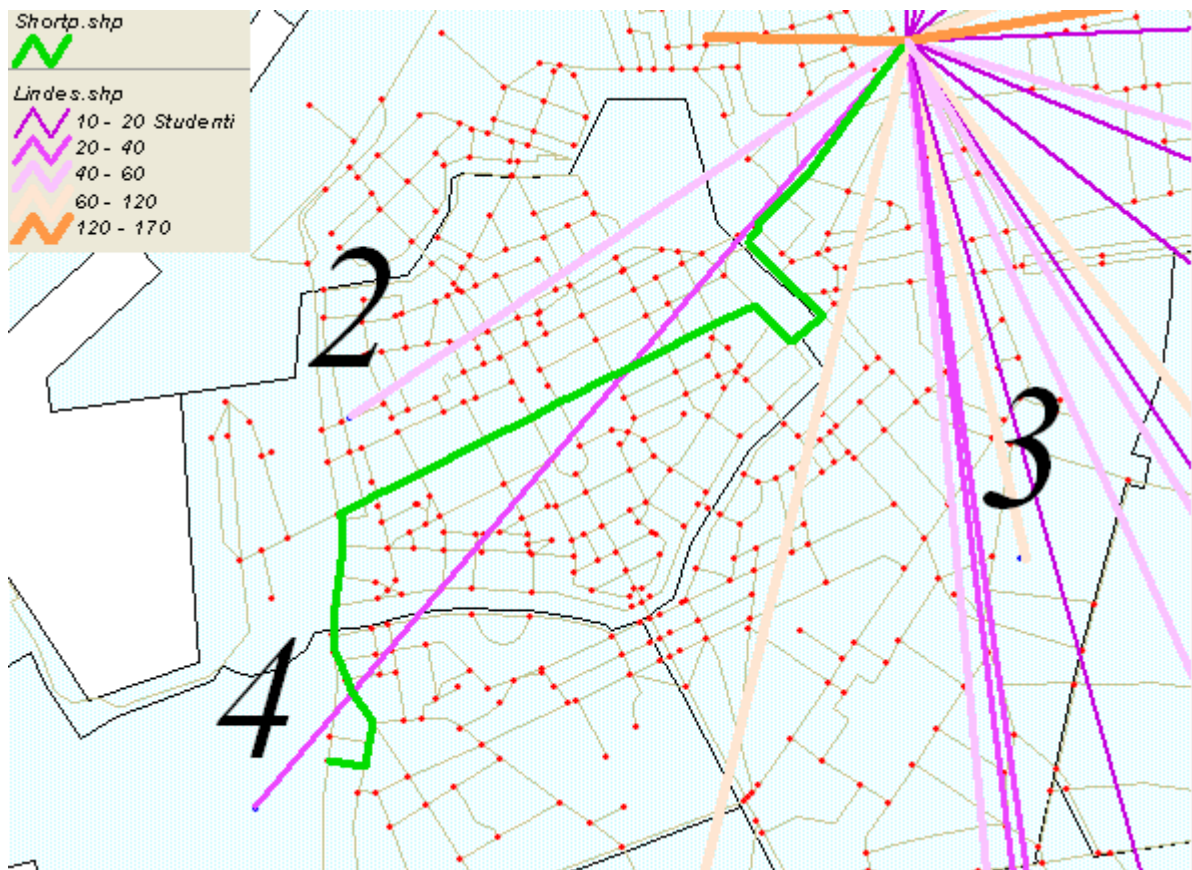


Figura 13 In verde, il cammino minimo tra la zona di traffico 4 e l'ITIS G. Galilei

6.2 Conclusioni

Dal risultato ottenuto, mostrato precedentemente, possiamo arrivare a delle conclusioni; analizzando prima le zone esterne al comune, possiamo vedere che:

- Il numero di frequentati provenienti dall'esterno è pari a 243, ovvero il 20,43% del totale.
- La macrozona est è quella che in generale origina il maggior numero di spostamenti, pari a 165, ovvero circa il 13,9% degli spostamenti. Questo fatto può essere prodotto dalla grande quantità di studenti che provengono dall'unico comune della provincia livornese che sta ad est della città, ovvero Collesalveti.
- La macrozona nord è quella che da origine al minor numero di spostamenti, pari a 15. Questo risultato probabilmente è dovuto alla presenza di altre scuole nella

zona nord, che comprende soprattutto il territorio della provincia di Pisa, che sono più vicine rispetto alla destinazione presa in considerazione in questo studio.

- La macrozona sud produce 63 spostamenti, pari a circa il 5,5% del totale.

Passando invece all'analisi degli spostamenti generati dalle zone di traffico, possiamo vedere che una grande parte degli spostamenti ha origine in zone adiacenti alla zona dell'ITIS G. Galilei: le zone di traffico 2, 3, 7, 8, 19 e 21 generano in totale 527 spostamenti, ovvero circa il 44% del totale. Osservando poi le altre zone, si può notare che mano a mano che ci si allontana dal nodo di destinazione, gli spostamenti sono minori.

Al di là dei risultati ottenuti dall'analisi dei dati, che può essere fatta in modo più approfondito ed accurato da entità di competenza, il lavoro aveva lo scopo principale di definire una metodologia da attuare per creare una sistema informativo per analizzare la domanda di mobilità.

Sono stati definiti in primo luogo dei metodi per la raccolta dei dati, descrivendo per ognuno le problematiche e le possibili soluzioni; è stata creata un'applicazione, in linguaggio VisualBasic, che ha permesso la normalizzazione degli indirizzi di residenza, con un basso margine di errore e incongruenza, descrivendo tutti i problemi legati a questa fase e le tecniche usate per la loro soluzione. È stata descritta la procedura per arrivare a definire la matrice Origine/Destinazione, contenente il centroide della zona di traffico di partenza ed il centroide relativo all'arrivo. Viene poi messo a disposizione dell'utente un sistema GIS, che prevede:

- L'aggregazione manuale (selezionando, graficamente, le zone Istat) oppure automatica (tramite una tabella di corrispondenze) in modo da formare zone di grandezza più significativa, le zone di traffico.
- Il trasferimento degli indirizzi di residenza, raccolti in precedenza, sul sistema GIS.
- La gestione della rete stradale (gestione dei nodi, degli archi, dei numeri civici, ecc).
- Un tool per il calcolo del cammino minimo tra due nodi.
- Il costante allineamento tra la banca dati geografica e la banca dati alfanumerica, tramite l'utilizzo di strumenti software quali ODBC e interfacciamento tramite linguaggio VisualBasic.
- Uno script che, tramite una matrice O/D, presenta graficamente le linee di desiderio, in modo da studiare ed analizzare la domanda di mobilità all'interno della città.

In definitiva, è stato progettato e realizzato un DSS, *Decision Support System*, che può essere messo a disposizione di autorità competenti, in questo caso, ad esempio, l'azienda ATL e le amministrazioni Comunale e Provinciale di Livorno, in modo da poter programmare e pianificare in modo corretto e coerente il servizio di trasporto pubblico urbano ed extraurbano.

7. RIFERIMENTI

- http://it.wikipedia.org/wiki/Decision_support_system
- http://it.wikipedia.org/wiki/Sistema_informativo_territoriale
- Cos'è il GIS (pdf), http://www.comune.argenta.fe.it/files/mappa_digitale/Cose_il_GIS.pdf
- <http://www.programmazione.it/index.php?entity=eitem&idItem=37371>
- <http://www.appuntisoftware.it/edit-distance-o-levenshtein-distance-un-algoritmo-di-similarita-tra-stringhe/>
- http://it.wikipedia.org/wiki/Distanza_di_Levenshtein
- <http://calogerosanfilippo.net/2008/01/13/programmazione-dinamica-distanza-di-levenshtein/>
- <http://alexandrerodichevski.chiappani.it/doc.php?n=223&lang=it>
- <http://santecaserio.altervista.org/wordpress/?p=51>
- www.club-cmmc.it/aziende/Procedura%20di%20normalizzazione.doc
- <http://www.uniserv.com/it/products/data-quality-functions-features/address-validation-data-capture.php>
- http://www.address-service-center.it/cms_vantaggi.html
- <http://it.wikipedia.org/wiki/CAPTCHA>
- <http://it.wikipedia.org/wiki/Bot>
- <http://php.html.it/articoli/leggi/2276/captcha-con-php/3/>
- <http://www.provincia.bergamo.it/Provpor/provBgViewEditorialNewsProcessWAl.jsp?editorialID=78762&nocache=1775029791>
- <http://www.vinzplanet.com/65/bloccare-gli-indirizzi-ip-dei-visitatori-fastidiosi/>
- <http://forum.it.altervista.org/php-mysql-e-apache-htaccess/101615-salvare-indirizzo-ip.html>
- <http://it.wikipedia.org/wiki/Shapefile>
- http://sit.provincia.brescia.it/cosa_banca_dati_geografica