



Università degli Studi di Padova

FACOLTA' DI INGEGNERIA

Corso di laurea in Ingegneria Informatica

**SVILUPPO DI APPLICAZIONI WEB PER
IL SISTEMA INFORMATIVO TERRITORIALE
DEL COMUNE DI FELTRE**

RELATORE: Prof. Massimo Rumor

LAUREANDO: Giuliano De Boni

Matr. 581190

Anno Accademico 2009/2010

INDICE

1.	INTRODUZIONE: TIROCINIO.....	3
2.	SISTEMI GIS	4
	2.1.INTRODUZIONE	4
	2.2.TIPOLOGIA DI DATI GEOGRAFICI	6
	2.2.1.DATI VETTORIALI	6
	2.2.2.DATI RASTER	6
	2.3.FUNZIONALITA'	7
	2.4.SOFTWARE.....	8
3.	SISTEMA GIS DEL COMUNE DI FELTRE: PIATTAFORMA PRECEDENTE	9
	3.1.DESCRIZIONE	9
	3.2.COMPONENTI TECNOLOGICHE	10
4.	SISTEMA GIS DEL COMUNE DI FELTRE: NUOVA PIATTAFORMA.....	16
	4.1.DESCRIZIONE	16
	4.2.COMPONENTI TECNOLOGICHE	19
5.	PROGETTO GESTIONE RACCOLTA RIFIUTI E DIFFERENZIATA	30
	5.1.OBIETTIVI.....	30
	5.2.DATI.....	31
	5.2.1.CARTOGRAFIA	32
	5.2.2.DATABASE	34
	5.2.2.1.SCHEMA E-R	34
	5.2.2.2.SCHEMA LOGICO	34
	5.2.2.3.CODICE SQL	35
	5.3.APPLICATIVO WEB	38
6.	CONSIDERAZIONI FINALI	46

1. INTRODUZIONE: TIROCINIO

Il tirocinio è stato svolto presso il CED del Comune di Feltre dal 8 marzo 2010 al 8 agosto 2010, per un totale di 500 ore.

L'argomento del tirocinio era il seguente: migrazione del sistema GIS comunale da un'architettura basata su software proprietari ad una basata su software open source.

La nuova architettura è stata sviluppata da una ditta esterna, ma delle operazioni di integrazione e migrazione se ne occupano i dipendenti del CED del Comune di Feltre.

In realta, il mio compito all'interno di questo progetto è stato di quello di progettare database e applicazioni web che si interfacciano con il sistema GIS, senza poter interagire direttamente con esso.

Il progetto che ho sviluppato riguarda la gestione dei dati delle isole ecologiche dislocate su tutto il territorio comunale, con progettazione del database gestionale e sviluppo dell'applicativo web per gestire tali dati, con la possibilità di collegarlo al sistema gis.

2. SISTEMI GIS

2.1. INTRODUZIONE

La conoscenza dell'ambiente e del territorio è diventata la base fondamentale su cui impostare le scelte di pianificazione territoriale dei vari enti pubblici.

Tale conoscenza del territorio si concretizza oggi con la realizzazione di un sistema di archivi geografici integrati (SIT, sistema informativo territoriale) e con l'utilizzo di strumenti informatici per la lettura, l'analisi e l'elaborazione dei dati stessi (*Geographic Information System, GIS*).

Ogni Sistema Informativo Territoriale (SIT) si propone, dunque, come strumento di supporto nella gestione dei processi decisionali, sia a livello pubblico che privato, poiché si pone come strumento di base per un'adeguata, efficace ed efficiente conoscenza del territorio.

Un sistema è un sistema informativo computerizzato che permette l'acquisizione, la registrazione, l'analisi, la visualizzazione e la restituzione di informazioni derivanti da dati geografici. Secondo la definizione di Burrough (1986) "il GIS è composto da una serie di strumenti software per acquisire, memorizzare, estrarre, trasformare e visualizzare dati spaziali dal mondo reale".



Esempio di un sistema GIS nel quale sono caricati livelli lineari e puntuali.

I sistemi GIS, come già detto, sono costituiti da procedure e strumenti informatici atti a rappresentare in maniera geoferenziata un oggetto che si trova in un determinato punto della Terra. La tecnologia GIS consente di integrare le operazioni tipiche dei database, quali la memorizzazione e l'immagazzinamento dei dati, le ricerche e le analisi statistiche, facilitando la lettura e l'interpretazione degli stessi attraverso la visualizzazione e l'analisi geografica forniti dagli strumenti di disegno cartografico, quindi sotto forma di mappe.

Il sistema viene utilizzato dalla quasi totalità degli utenti pubblici e da molti privati, sia nella gestione dati in tempo reale (urbanizzazione, reti tecnologiche, traffico, navigazione, ecc.), sia per lo studio di progetti, previsioni di scenari, definizione di strategie e di pianificazioni a lungo

termine come i Piani Regolatori nei loro vari livelli (Comunali, Comprensoriali, Provinciali, Regionali, Statali).

Il GIS offre la possibilità di filtrare le varie informazioni e di poter evidenziare poi le stesse a vari livelli di visualizzazione, sia in termini di scale mappali, sia in termini di colorazione, offrendo una veloce percezione di ciò che si vuole rappresentare o cercare.

2.2. TIPOLOGIA DI DATI GEOGRAFICI

L'informazione territoriale può essere codificata in un sistema informativo geografico attraverso due tipologie principali di dato: il dato vettoriale e il dato raster.

2.2.1. DATI VETTORIALI

I dati vettoriali sono costituiti da elementi semplici quali punti, linee e poligoni, codificati e memorizzati sulla base delle loro coordinate. Un punto viene individuato in un sistema informativo geografico attraverso le sue coordinate reali ($x1$, $y1$); una linea o un poligono attraverso la posizione dei suoi nodi ($x1$, $y1$; $x2$, $y2$; ...). A ciascun elemento è associato un record del database informativo che contiene tutti gli attributi dell'oggetto rappresentato.

I dati, all'interno di un GIS, sono memorizzati secondo diverse strutture dati, spesso proprietarie; allo scopo di permettere il dialogo ed il trasferimento dei dati tra i diversi sistemi sono stati definiti diversi formati di trasferimento. Tipici dati memorizzati in formato vettoriale sono quelli che provengono dalla digitalizzazione manuale di mappe, dai rilievi topografici con strumenti di campagna, dai CAD, dai GPS (Global Positioning Systems).

2.2.2. DATI RASTER

Tipici dati raster invece sono quelli generati dagli scanner e da sensori come quelli utilizzati per le immagini da satellite.

Il dato raster permette di rappresentare il mondo reale attraverso una matrice di celle, generalmente di forma quadrata o rettangolare, dette pixel. A ciascun pixel sono associate le informazioni relative a ciò che esso rappresenta sul territorio. La dimensione del pixel (detta anche *pixel size*), generalmente espressa nell'unità di misura della carta (metri, chilometri etc.), è strettamente relazionata alla precisione del dato.

Per registrazione si intendono le tecniche necessarie a georeferenziare e raddrizzare le immagini raster. Infatti le foto aeree e le immagini da satellite, oltre a dover essere posizionate correttamente (facendo collimare le coordinate dei punti noti a terra con quelle degli oggetti presenti nell'immagine), devono anche essere ortogonalizzate cioè ricalcolate tenendo conto dell'angolo da cui sono state prese. Oggi sono disponibili programmi che, oltre a fare questo, garantendo così la possibilità di visualizzare, in sovrapposizione, immagini raster con i corrispondenti dati vettoriali, sono anche in grado di interpretare le immagini identificando e visualizzando ad esempio i pixel in base ai parametri qualitativi misurati nelle varie bande (umidità e temperatura al suolo, ecc.).

Nel modello dei dati di un GIS, i dati vettoriali e i dati raster coesistono e si integrano a vicenda e sono generalmente usati i primi per dati discreti e i secondi per dati continui (ad esempio rete viaria vettoriale derivata dalla cartografia per i primi e umidità al suolo derivata da immagini da satellite per i secondi). Inoltre sono disponibili programmi in grado di convertire in modo più o meno automatico dati raster in vettoriali e viceversa. Entrambi i tipi di dati possono essere associati ad attributi: per i primi saranno legati alle primitive grafiche e agli oggetti, per i secondi ai singoli pixel.

2.3. FUNZIONALITA'

Un sistema 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. Il GIS 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.

I GIS presentano normalmente delle funzionalità di analisi spaziale ovvero di trasformazione ed elaborazione degli elementi geografici e degli attributi e funzionalità di rappresentazione e di aggiornamento delle mappe.

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 sulle 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

2.4. SOFTWARE

Alcuni software GIS open source:

- GRASS GIS
- Quantum GIS
- gvSIG
- Kosmo
- OpenJUMP
- uDig

Alcuni software GIS proprietari:

- ArcGIS
- Geomedia
- Fotorad GIS
- MapINFO
- Autodesk MAP GUIDE

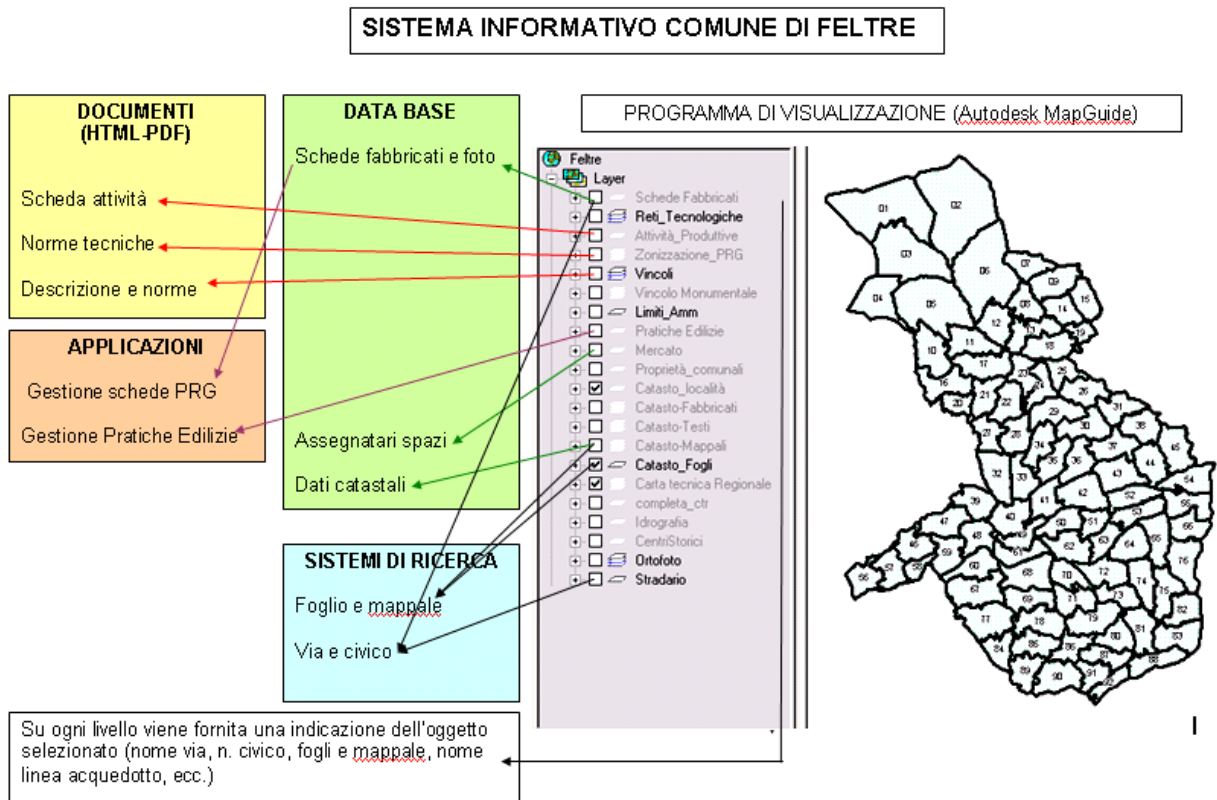
3. SISTEMA GIS DEL COMUNE DI FELTRE: PIATTAFORMA PRECEDENTE

3.1. DESCRIZIONE

Il sistema informativo territoriale del Comune di Feltre è stato avviato dall'Ufficio Urbanistica, che ha al suo interno un settore cartografico, in quanto era stata rilevata la necessità di collegare ad ogni edificio situato sul territorio comunale tutte le informazioni possibili (pratiche edilizie, catasto, vincoli, norme PRG, immagini, dati urbanistici, ecc.) al fine di fornire una corretta informazione ai cittadini ed ai tecnici.

Successivamente, sfruttando la proprietà fondamentale di un sistema informativo territoriale, la “georeferenziazione” (posizionamento di oggetti su una mappa in base alle coordinate geografiche) è stato possibile collegare tra loro informazioni presenti in diverse banche dati (viabilità, prg, catasto, anagrafe, ecc.), oltre a quanto veniva già rappresentato. Partendo da una documentazione cartografica del territorio del Comune di Feltre (CTR, Catasto, Ortofoto, ecc.) sono state aggiunte informazioni più dettagliate (Viabilità, Acquedotto, P.R.G., Vincoli, Civici, Proprietà, ecc.) alle quali sono state associate, di volta in volta, informazioni già possedute dagli uffici.

Questo processo ha consentito la visualizzazione dei dati su mappe create con il sistema GIS e comprendeva anche altre funzionalità come, ad esempio, il posizionamento sulla mappa mediante la ricerca di un dato specifico (via, civico, ecc) e attraverso quel dato la possibilità di ottenere eventuali informazioni aggiuntive.



3.2. COMPONENTI TECNOLOGICHE

L'architettura di un sistema GIS si basava solitamente sul modello client-server. In questo caso il software GIS veniva gestito da un server che si occupava di elaborare e rendere disponibili le informazioni, mentre il client accedeva al server tramite un'interfaccia per ottenere i dati e svolgere le opportune operazioni. Il server quindi conteneva i database che memorizzavano le informazioni, i componenti software (DBMS) che si occupavano di gestire le banche dati, l'ambiente GIS e altre componenti che fornivano ulteriori servizi. Il client invece utilizzava un pacchetto applicativo che gli consentiva di interagire con il server.

Negli ultimi anni lo sviluppo di Internet ha permesso di creare un'architettura client-server basata sull'utilizzo del web per accedere ai sistemi GIS. In particolare, nel nostro caso, a livello client si utilizza un browser per lavorare e gestire l'intero sistema anche se alcuni client utilizzano anche pacchetti software proprietari specifici.

La piattaforma utilizzata precedentemente dal Comune di Feltre si basa su un'architettura proprietaria e utilizza software prodotti dalle aziende Microsoft e Autodesk.

L'architettura di un sistema GIS si può dividere in tre livelli:

- Livello dati.

Si tratta della struttura dati in cui vengono memorizzate le informazioni di tipo cartografico e le informazioni aggiuntive come attributi o altro.

Nella vecchia architettura le informazioni venivano memorizzate su database formato access (.mdb).

- Livello applicativo.

Il processo che si occupa di generare le mappe a partire dalle informazioni cartografiche memorizzate nel database.

I software utilizzati sono prodotti dalla Autodesk:

- **Autodesk MapGuide Author:**

Autodesk MapGuide Author consente di creare le mappe mediante la configurazione dei file di visualizzazione della mappa stessa (formato file MWF).

Ogni file MWF contiene le specifiche di visualizzazione della mappa. Queste specifiche comprendono i contorni della mappa, il colore di sfondo, le informazioni sulla fonte dei dati (database SQL, file di dati spaziali e file raster), i layer dei dati cartografici (ad esempio, strade e stati), le informazioni sulla visualizzazione dei layer (ad esempio, attributi e scale visibili), nonché le specifiche di configurazione per la visualizzazione della mappa (ad esempio, menu e legende).

È possibile quindi incorporare uno o più file MWF, oppure creare un collegamento ad essi, in una pagina Web. Quando un utente apre la pagina Web o fa clic sul collegamento nel browser, viene caricato Autodesk MapGuide Viewer e visualizzata la mappa.

- **Server WEB:**

Il software del server Web elabora e invia vari tipi di file, come pagine Web e file di visualizzazione della mappa (MWF), in risposta alle richieste inviate da altri computer. Quando l'utente richiede i dati di una mappa, la richiesta viene ricevuta dal server Web e inviata ad Autodesk MapGuide Server. Autodesk MapGuide supporta due tipi di server Web: Microsoft Internet Information Server (IIS) 4.0 o successivo tramite l'interfaccia standard CGI e ISAPI e Netscape Enterprise Server 3.51 tramite l'interfaccia standard CGI e NSAPI.

Il server WEB utilizzato dal Comune di Feltre è il Microsoft Internet Information Server (IIS).

- **Autodesk MapGuide Server:**

Questo software fornisce i dati della mappa in risposta alle richieste inviate da Autodesk MapGuide Author e Autodesk MapGuide Viewer quando un utente visualizza una mappa utilizzando tali programmi.

Quando si effettua una richiesta, Autodesk MapGuide Server legge la richiesta per determinare i dati da fornire, quindi invia i dati dalla fonte dei dati spaziali, dai file raster o dalle fonti dei dati OLE DB, in base alle specifiche effettuate nella richiesta. Autodesk MapGuide Server consente il controllo dell'accesso alle fonti dei dati mediante il controllo delle password, degli ID utente e altre impostazioni opzionali.

- **Livello client.**

- **Il browser Web** è un'applicazione client che legge i dati ipertestuali sotto forma di pagine HTML contenenti testo, grafica e codice.

Generalmente, le pagine Web vengono inviate su Internet, intranet o extranet. Una rete Intranet utilizza la tecnologia Internet, ma può essere usata solo all'interno di una società o di un'organizzazione; le informazioni presenti su una rete Intranet non sono disponibili ad un utente esterno. Una rete extranet permette l'accesso ad una rete intranet ad utenti esterni autorizzati, ad esempio clienti che accedono ad un sito tramite password.

Oltre alla lettura e alla visualizzazione di pagine Web, un browser Web permette anche il trasferimento di file tramite il protocollo di trasferimento file (FTP) e l'esecuzione di programmi ausiliari come Autodesk MapGuide Viewer, che verrà descritto di seguito.

- **Autodesk MapGuide Viewer:**

Autodesk MapGuide Viewer visualizza una mappa nel browser Web quando si apre una pagina Web contenente un file MWF incorporato o viene aperto un collegamento ad un file MWF.

Autodesk MapGuide Viewer è disponibile in tre versioni:

- La versione Plug-In per l'uso con Netscape Navigator su sistemi operativi Windows.
- La versione ActiveX Control per l'uso con Microsoft Internet Explorer su sistemi operativi Windows.
- Java Edition per l'uso su sistemi operativi Sun Solaris e Apple Macintosh.

Il client utilizza, per accedere al sistema GIS, il browser web Internet Explorer prodotto della Microsoft Corporation e la versione ActiveX Control del software Autodesk MapGuide Viewer.

http://feltre_1lpp/sit/webmap/gis2schede.asp - Microsoft Internet Explorer

File Modifica Visualizza Preferiti Strumenti ?

Scheda Edificio

Ecografico: 1002500150000 **Località:** FOEN
Via: VIA VALLINA **Civico:** 15
Foglio: 28 **Mappale:** 236
Epoca: REALIZZATO TRA IL CAT.AUSTRIACO E IL 1945 **Utilizzo:** UTILIZZATO
Tipologia: IN CORTINA
Destinazione Uso: RESIDENZA **Cond Fische:** 3-SUFFICIENTE
N.Piani: 3 **Sup.Coperta:** 142 **Volume:** 1278

Descrizione: EDIFICIO DI TESTA DI UNA SCHIERA EDILIZIA SVILUPPATA LUNGO VIA MADONNETTA DI FOEN. A PIANO TERRA SU VIA VALLINA SI TROVA UNA ATTIVITA' COMMERCIALE, AL PIANO SUPERIORE UNA RESIDENZA. IL RETRO DELL'EDIFICIO PRESENTA UNA RAMPA DI ACCESSO AL GARAGE SEMINTERRATO CON SBOCO SU VIA MADONNETTA, DALLA STESSA RAMPA SI ACCEDE LA CIV. 1 DI TALE VIA. QUESTA PORZIONE DI EDIFICIO RISULTA PROPRIETA' UNITARIA DA TERRA A CIELO CON DESTINAZIONE RESIDENZIALE.

Note:

C:\Sit\WebMap\FotoSchedeB\10025001500001.jpg

Operazione completata Risorse del computer

autodesk

Schede Edifici : 15

1 'Schede Edifici' selezionati 1 : 1.620

Intranet locale

GPE versione 2

Finestre Pratica Modifica Stampe Tabelle Strumenti Configura

Menu 1

Apri pratica

Apri per: Codice ecografico

10410007000

Pratica	Oggetto
2001/0043	RESTAURO E RISTRUTTURAZIONE FABBRICATO
2003/0714	VARIANTI ALLA C.E. NR. 219/01 PER MODIFICHE FABBRICA

http://feltre_1lpp/SIT/WebMap/feltre.mwf - Microsoft Internet Explorer

File Modifica Visualizza Preferiti Strumenti ?

Indirizzo http://feltre_1lpp/SIT/WebMap/feltre.mwf

autodesk

Cittadella

10410007000

Pratiche Edilizie : 10410007000

1 : 1.216

Intranet locale

12/03/2004 9.16

4. SISTEMA GIS DEL COMUNE DI FELTRE: NUOVA PIATTAFORMA

4.1. DESCRIZIONE

Alla fine del 2008 è stata installata la piattaforma GIS denominata “InformCity”, realizzata dalla ditta Infracom IT e si è proceduto alla conversione di alcuni layer cartografici.

Il processo di conversione dalla piattaforma precedente a quella nuova è iniziato con l’installazione di “InformCity” e sta per essere ultimato in questo periodo, con la conversione degli ultimi database e layer cartografici.

In questi due anni sono state installate e utilizzate varie versioni del nuovo sistema.

I motivi che hanno portato alla sostituzione del sistema precedente, basato su architettura proprietaria (Autodesk), sono molteplici e i principali sono descritti di seguito, tra le caratteristiche del nuovo sistema **“Informcity”**.

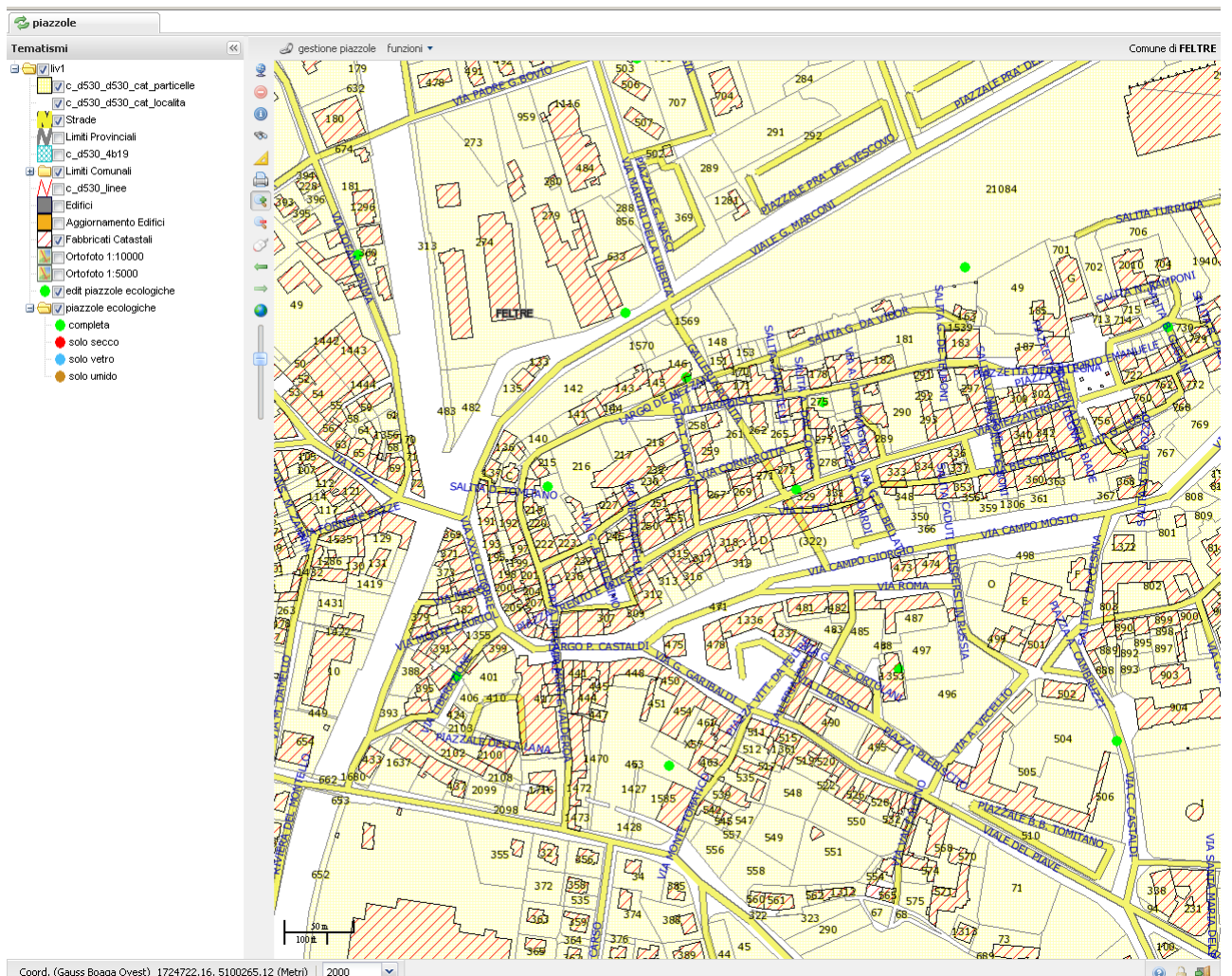
Caratteristiche generali di “Informcity”

“Informcity” è un Sistema Informativo Territoriale via Web (WebGIS) sviluppato con l’utilizzo di tecnologie OpenSource (MapServer, PostgreSQL e PostGIS), che consente di affrontare in modo integrato aspetti territoriali ed urbanistici per la gestione del territorio.

Le caratteristiche principali sono:

- Gestione uniforme e centralizzata dei dati attraverso l’utilizzo di un database relazionale con estensione spaziale;
- La presenza di funzioni per la sovrapposizione, l’interrogazione dinamica di diversi tipi di cartografia e archivi alfanumerici e l’analisi spaziale.
- La costituzione di cartografie personalizzate.

- L'accesso al sistema è consentito solo tramite autenticazione (login, password). Per consentire la massima flessibilità di accesso alle diverse funzioni/dati da parte di ciascun utente, è previsto un sistema di relazioni dinamiche, fra utenti, ruoli e funzioni/dati, con una gestione delle autorizzazioni e delle gerarchie dei permessi;
- La consultazione delle banche dati cartografiche avviene grazie a funzioni di misura (visualizzazione della misura di perimetri ed aree disegnate sulla mappa dall'utente) e stampa (in formato .pdf) con possibilità di scelta del formato pagina (A3/A4), della risoluzione (alta/bassa), della scala e d'inserimento del titolo. Questo consente ai tecnici ed ai singoli utenti di cercare, trovare e riportare su supporto cartaceo le informazioni desiderate.
- L'interrogazione delle banche dati cartografiche ed alfanumeriche utilizzando funzioni di query basate sulle classi di tipologia di uso del suolo, copertura vegetale, indirizzo e/o del numero civico per la parte urbanizzata, ecc. - con visualizzazione sulla mappa e ricerca cartografica (interrogazioni banche dati cartografiche e link alle rispettive banche dati alfanumeriche eventualmente collegate).
- La possibilità di scaricare dalla propria postazione i dati alfanumerici risultanti dall'interrogazione.
- Wizard che permette e facilita il collegamento di database alfanumerici (WebServices, Oracle, MySql, ODBC, Access, PostgreSQL)
- Assenza di vincoli legati a licenze software: ciascun utente può accedere all'applicazione e ai dati in essa gestiti liberamente (secondo le autorizzazioni previste), è sufficiente un Browser per la visualizzazione dei dati, senza la necessità di installare alcun plugin.
- Possibilità di offrire condivisione delle informazioni cartografiche tramite i protocolli di comunicazione WMS – WFS.
- Infine per quanto riguarda la Gestione del Territorio del Comune di Feltre: possibilità di effettuare sovrapposizioni e interrogazioni dinamiche di diversi tipi di cartografia (foto aeree, PRG, CRT, cartografie tematiche, etc.) e di database alfanumerici.



4.2. COMPONENTI TECNOLOGICHE

L'architettura di "Informcity" si basa sul modello client-server e si può suddividere in tre livelli:

- Livello dati:

il database utilizzato è PostgreSQL con l'estensione spaziale PostGIS.

PostGIS è un'estensione del database relazionale PostgreSQL per l'archivio e la gestione dei dati geografici; in pratica si tratta di una libreria pensata e studiata apposta per PostgreSQL. E' un database che segue lo standard OGC e consente l'interoperabilità tra applicazioni desktop, analitiche e Web nel campo del GIS. Inoltre permette l'accesso multiutente a grandi moli di dati, sia geografici che alfanumerici e la gestione delle relazioni tra di essi, garantendone l'integrità. È prevista anche la gestione dei dati tridimensionali e si incammina verso quella dei raster.

Lo schema classico delle applicazioni desktop GIS era solitamente basato su un unico sistema sia per la visualizzazione dei dati che per la relativa archiviazione su file. Uno dei modelli più utilizzati (ad esempio il formato ESRI Shape file) era quello in cui i dati vengono divisi in tre parti (tre file distinti): una tabella per le geometrie (ad esempio file .shp), una per gli attributi (ad esempio file .dbf) ed una per gli indici delle relazioni fra le prime due (ad esempio file .shx).

Questo tipo di schema era utilizzato dai primi sistemi GIS per risolvere scenari d'uso in cui l'interazione con l'applicazione era affidata ad un unico utente ma, quando gli utenti che dovevano accedere alla base dati erano più di uno, è stato necessario introdurre l'utilizzo di sistemi di database più evoluti quali i DBMS e PostgreSQL è uno di questi.

L'interfaccia fondamentale al database è il linguaggio SQL (Structured Query Language), un linguaggio standard, che consente di effettuare operazioni di selezione, inserimento, modifica, calcolo, applicazione di funzioni, ecc., su un database costituito da tabelle in relazione tra loro (database relazionale, spesso si usa il nome relazione per indicare una tabella).

L'architettura di un database geografico si distingue da quella dei database classici per il tipo di dato che deve gestire: il dato geografico. Pertanto, la prima cosa richiesta ad un DBMS che volesse andare in questa direzione era una struttura in grado di gestire questa tipologia di dati. Era necessario, non solo definire la struttura del dato geografico, ma anche arricchire il linguaggio SQL con funzionalità in grado di calcolare la distanza fra due punti o l'area di un poligono su uno spazio bidimensionale e/o tridimensionale, oppure di effettuare la conversione da un sistema di riferimento ad un altro, ecc.

L'Open Geospatial Consortium (precedentemente chiamato Open GIS Consortium; ha elaborato, grazie alla collaborazione di 5 grandi aziende: due nel settore GIS (ESRI e MalInfo) e tre in quello database (IBM, Informix e Oracle), un documento relativo agli standard per i formati dei dati geografici, pubblicato nel maggio del 1999.

Il documento dell'OGC, citato in precedenza definisce:

- Quali tabelle debbano essere sempre presenti in un DBMS con estensioni geografiche (tabella delle geometrie e tabella dei sistemi di riferimento);
- La tipologia di geometrie da archiviare denominate Simple Features (point, line, polygon, ecc.) offrendo due possibili schemi: uno testuale (che occupa maggior spazio ma di facile lettura) denominato (Well Know Text); uno binario (con caratteristiche opposte al primo) denominato WKB (Well Know Binary);
- L'insieme delle istruzioni SQL per l'analisi spaziale.

Questo documento è divenuto il punto di riferimento per le estensioni delle funzioni SQL Spatial di qualsiasi DBMS.

Le caratteristiche principali di PostgreSQL:

- E' in grado di supportare un numero elevato di tipi di dati binari e testuali.
- Permette di gestire database molto grandi anche su sistemi operativi con capacità limitate.
- Include la possibilità di utilizzare funzioni che permettono la replicazione dei dati, load balancing, uso di multiprocessori e cluster.
- Supporta diverse modalità di autenticazione crittografate.

Una volta applicate le estensioni PostGis ad un database PostgreSQL viene definita una struttura dati per l'archiviazione di dati spaziali secondo le specifiche Simple Features.

Queste specifiche permettono l'archiviazione di 7 tipi diversi di dato:

POINT (punto), LINESTRING (linea), POLYGON (poligono), MULTYPOINT (collezione dei punti), MULTILINESTRING (collezione di linee), MULTYPOLYGON (collezione di poligoni), GEOMETRYCOLLECTION (collezione di geometrie di vario tipo).

Ogni tipo ha una propria sintassi che consente l'archiviazione delle coordinate sia su piani bidimensionali (quindi x e y) sia su piani tridimensionali (x, y e l'altitudine z).

Per esempio, nel caso di un POINT:

coordinate piano 2D POINT (0 0)

Coordinate piano 3D POINT (0 0 0)

Ogni record di dati viene arricchito dal “prefisso”SRID (Spatial Reference ID), variabile che indica il sistema di riferimento nel quale sono archiviate le geometrie. Il valore attribuito a SRID è scelto tra quelli presenti nella tabella spatial_ref_sys, creata di default in ogni database geografico. In caso di SRID ignoto si usa l’indice -1.

Esempio di archiviazione delle coordinate di un punto su un piano bidimensionale secondo il sistema di riferimento Gauss-Boaga ovest:

SRID=26591, POINT (2455,443343 56565,32).

L’inserimento dei dati geografici può avvenire secondo due modalità:

- attraverso l’uso di comandi SQL secondo la sintassi Simple Features;
- importazione dei dati attraverso tool testuali o grafici.

Nel caso in cui si utilizzi direttamente un’istruzione SQL è necessario creare una tabella secondo la sintassi classica.

Esempio:

```
CREATE TABLE punti (ID int4, DESCRIZIONE varchar (40)).
```

Successivamente bisogna aggiungere a questa tabella il campo corrispondente alla colonna dei dati geografici secondo questa sintassi:

```
SELECT AddGeometryColumn  
(‘nome_tabella’, ‘colonna_geometrie’, srid, ‘tipo_geometria’,  
dimensione);
```

dove:

nome_tabella = nome della tabella a cui applicare le geometrie;

colonna_geometrie = nome del campo che dovrà contenere il dato geometrico;

srid = id della tabella con i sistemi di riferimento (se ignoto:-1);

tipo_geometria = uno dei 7 valori di simple features (POINT, LINESTRING, ecc.);

dimensione = 2 per le geometrie bidimensionali, 3 per quelle tridimensionali.

L'inserimento delle singole geometrie avviene poi, secondo le classiche istruzioni SQL.

Esempio:

```
INSERT INTO nome_tabella (colonna_geometrie, descrizione) values  
( 'SRID=-1, POINT (0 0 0) 'Punto inserito').
```

In alternativa alle istruzioni SQL, PostGIS offre il tool di importazione di file in formato ESRI shapelite “shp2pgsql” che crea in automatico tutto il codice SQL necessario. In questo caso, se non definita dall'utente, la colonna con le geometrie prende di default il nome “the_geom”.

Le funzionalità PostGIS sono classificabili in categorie:

- Funzioni di base.
Permettono la creazione/eliminazione di colonne geometriche e l'attribuzione dei dati ad un determinato sistema di riferimento.
- Funzioni di relazioni fra geometrie.
Calcolo della distanza fra due geometrie, verifica della eventuale sovrapposizione, intersezione, inclusione, ecc. tra forme geografiche distinte.
- Funzioni di calcolo sulle geometrie.
Consentono di calcolare area, perimetro, buffer, centroide, ecc., di una data geometrica.
- Funzione di “informazioni” sulle geometrie.
Attraverso di esse è possibile conoscere il tipo di geometria presente in un dato campo, l'ID del sistema di riferimento utilizzato, le coordinate dell'ultimo punto di una geometria, il numero di punti contenuti, il valore x o y o z di un dato record, ecc..
- Funzioni di creazioni di geometrie.
Permettono di creare dati geometrici e di archivarli in formato standard OGC a partire da un insieme di coordinate. Questo permette di trasformare in un database geografico un database alfanumerico in cui le

coordinate dei punti siano, ad esempio, archiviate in due campi 'x' e 'y'.

Oltre alle funzioni sopra descritte, e che fanno parte di quelle standard OpenGIS, PostGIS ha numerose funzioni avanzate. Le più importanti sono elencate di seguito:

- Funzioni di calcolo di misure.
Permettono di calcolare le aree, i perimetri, le lunghezze, gli sferoidi, ecc., in relazione al sistema di riferimento e al tipo di piano (bidimensionale o tridimensionale).
- Funzioni di modifica delle geometrie.
Consentono di manipolare le geometrie effettuando ad esempio operazioni di semplificazione, di traslazione, di inversione, di conversione da un sistema di riferimento ad un altro, ecc.

- Livello applicativo.

L'applicativo server utilizzato è il server geospaziale Mapserver.

MapServer è un ambiente di sviluppo OpenSource per la visualizzazione e la consultazione di dati provenienti da sistemi GIS attraverso Internet. MapServer supporta le specifiche web dell'Open Geospatial Consortium, già descritte in precedenza.

E' stato inizialmente sviluppato dalla University of Minnesota (UMN) nell'ambito del progetto ForNet sostenuto dalla NASA insieme al Minnesota Department of Natural Resources. Successivamente il suo sviluppo è continuato con un altro progetto della NASA: il progetto TerraSip. Attualmente MapServer è in continua crescita, diffuso in ogni parte del mondo da un numero sempre crescente di sviluppatori e supportato da diversi gruppi ed organizzazioni che ne consentono una costante evoluzione.

E' un sistema multiplatforma in grado di interfacciarsi con i più conosciuti database relazionali. Supporta vari linguaggi di programmazione nell'utilizzo delle API scritte in linguaggio 'C' e consente l'utilizzo di diversi formati cartografici vettoriali e raster.

Il motore principale è un programma CGI (Common Gateway Interface) che viene eseguito all'interno di un Web Server e che utilizza vari file di configurazione predisposti in appositi formati (Map File, Template, ecc).

Il suo funzionamento è il seguente:

il server riceve una richiesta contenente come parametro, inviato tramite l'indirizzo URL, il nome del file di configurazione di una mappa ed eventuali altre preferenze. MapServer risponde inviando direttamente l'immagine della mappa o copiando il file sul sistema locale e inviando in risposta al cliente un file HTML contenente le principali informazioni sulla mappa prodotta (indirizzo URL della carta, nuovo extent , nuova scala, etc.)

La visualizzazione di una mappa con MapServer si basa su tre elementi:

- il file di configurazione della carta richiesta
- i layer contenuti in formati vari
- eventuali modifiche inviate via l'URL di chiamata

Il file di configurazione è un semplice file di testo contenente delle parole chiave che permettono di identificare gli elementi di configurazione della mappa.

Il seguente è un esempio di file di configurazione che definisce una carta contenente un solo layer di tipo linea chiamato "fiumi".

MAP

NAME prov_ts

STATUS ON

SIZE 600 600

EXTENT 2409000.00 5048000.00 2436000.00 5075000.00

SHAPEPATH "../data"

UNITS meters

```

WEB
TEMPLATE "../templates/all.xml"
IMAGEPATH "C:\progetti\schede\schede\schedeWeb\tempImages\"
IMAGEURL "tempImages/"
END # END_WEB

LAYER
NAME fiumi
STATUS ON
DATA fiumi
TYPE LINE
CLASS
SYMBOL 0
NAME "fiumi"
COLOR 0 0 0
SIZE 2
END # END_CLASS
END # END_LAYER
END

```

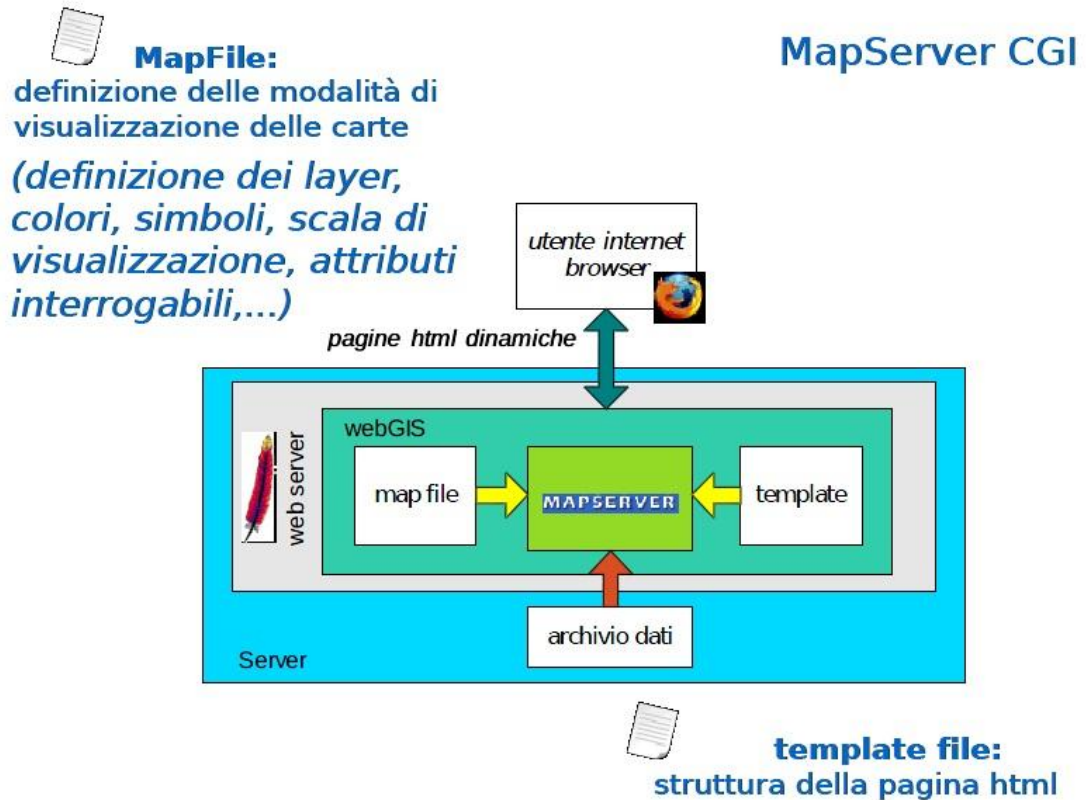
Le applicazioni basate su MapServer necessitano dei seguenti componenti:

1. il software MapServer;
2. un server HTTP (Apache, IIS o altri);
3. un file di configurazione (Mapfile) che controlli il modo in cui MapServer gestisce i dati;
4. un file di template (modello) per il controllo dell'interfaccia utente nella finestra del browser;
5. un insieme di dati GIS.

Esiste un altro approccio all'utilizzo di Mapserver e si basa sull'impiego di linguaggi di script per la generazione di pagine HTML dinamiche e accesso ai servizi di Mapserver attraverso le chiamate alle API.

Il server utilizzato è il server Apache.

Schema riassuntivo del funzionamento di Mapserver:



- Livello client:

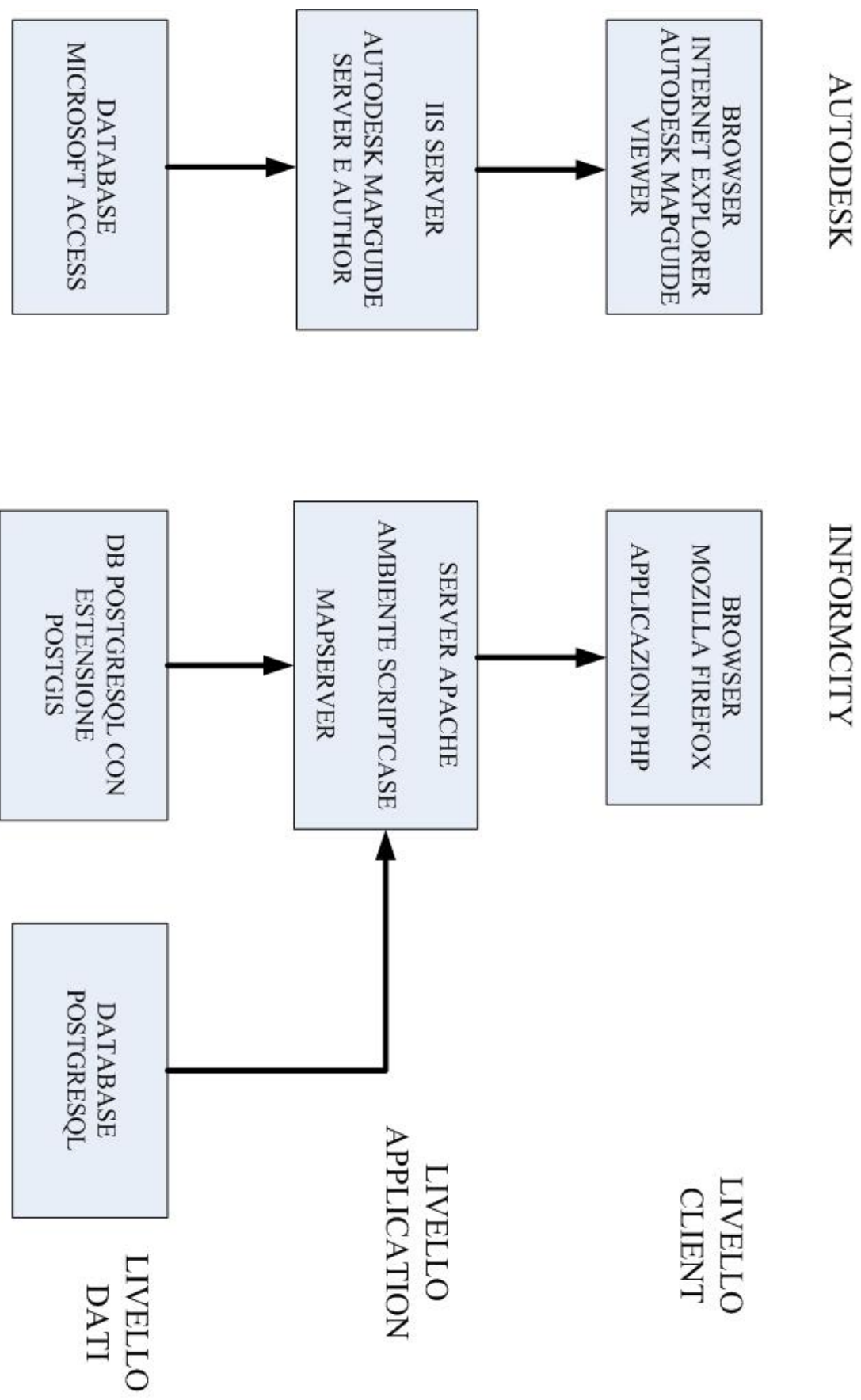
Il server WEB utilizzato dal sistema GIS “Informcity” è, come già detto in precedenza, il server Apache e il linguaggio utilizzato per lo sviluppo di applicazioni WEB è PHP.

In particolare è stato scelto l’ambiente di sviluppo Scriptcase che permette di realizzare applicazioni in PHP basate sui principali RDBMS sul mercato in modo praticamente automatico.

A livello client per interfacciarsi al sistema viene utilizzato il browser Mozilla Firefox (aggiornato all’ultima versione). Il sistema funziona anche con i principali browser in circolazione (Internet Explorer, Google Chrome, Opera) ma la scelta è ricaduta su Firefox in quanto è uno dei browser migliori sotto molti punti vista (sicurezza, usabilità ecc) ed è open source.

Ora di seguito viene proposto uno schema riassuntivo che confronta le due architetture presentate in precedenza.

CONFRONTO TRA LE DUE PIATTAFORME DEL COMUNE DI FELTRE



5. PROGETTO GESTIONE RACCOLTA RIFIUTI E DIFFERENZIATA

5.1. OBIETTIVI E REQUISITI

Il Comune di Feltre ha introdotto, a partire da maggio 2010, un nuovo sistema per la raccolta dei rifiuti (RSU) e della raccolta differenziata. L'ufficio Ambiente del Comune ha rilevato la necessità di gestire i dati riguardanti le piazzole della raccolta dei rifiuti e della raccolta differenziata che sono dislocate sul territorio comunale. Inoltre è necessario realizzare anche una mappa di tutte le piazzole presenti sul territorio utilizzando il sistema GIS comunale; infine si vuole collegare a questa mappa un gestionale web per memorizzare tutti i dati necessari alla gestione del sistema di raccolta dei rifiuti.

Il mio compito è stato quello di progettare una base di dati per gestire tali informazioni e sviluppare su questa un applicativo web, utilizzando l'ambiente di sviluppo Scriptcase, che verrà appunto collegato al sistema GIS.

REQUISITI APPLICATIVO WEB

La procedura scelta per effettuare l'inserimento dei dati è la seguente:

- Inserimento dei dati nel sistema GIS, disegnando sulla mappa le piazzole nelle posizioni dove si trovano.
- Sincronizzazione dei dati cartografici (le piazzole disegnate sulla mappa) con i dati dell'applicativo gestionale che gestisce le informazioni riguardanti le piazzole. L'applicativo dovrà scorrere il database cartografico per cercare le nuove piazzole inserite nel sistema GIS e quindi dovrà creare il relativo record sul database gestionale.
- Inserimento dei dati delle piazzole nel gestionale.

REQUISITI DATABASE

L'applicativo WEB che si vuole sviluppare deve gestire i dati che riguardano le piazzole ecologiche dislocate sul territorio comunale.

In particolare, per ogni **piazzola** è necessario memorizzare un codice 'id_piazzola' con valore numerico auto incrementante che la identifica univocamente, la posizione dove si trova indicando la via ed eventualmente il numero civico, i dati catastali (foglio e mappale), la tipologia della piazzola (completa, solo umido, solo secco ecc), una breve descrizione, l'eventuale proprietario, se la piazzola non sorge su suolo pubblico; infine è necessario prevedere anche due campi che permettano di tener traccia di eventuali note e autorizzazioni.

Per ogni **cassonetto** si vuole memorizzare un codice 'id_cassonetto' con valore numerico auto incrementante che lo identifica univocamente, la piazzola dove è collocato il contenitore, il tipo di cassonetto, il volume di rifiuti che può contenere, il numero della calotta, il numero di svuotamenti settimanali, la data di collocazione nella relativa piazzola e il numero d'inventario.

Per ogni **tipologia di contenitore** è necessario memorizzare il nome che la identifica univocamente, il volume e il fornitore.

Infine per **ogni tipologia di piazzola** è necessario memorizzare un codice numerico auto incrementante che la identifica univocamente e il tipo.

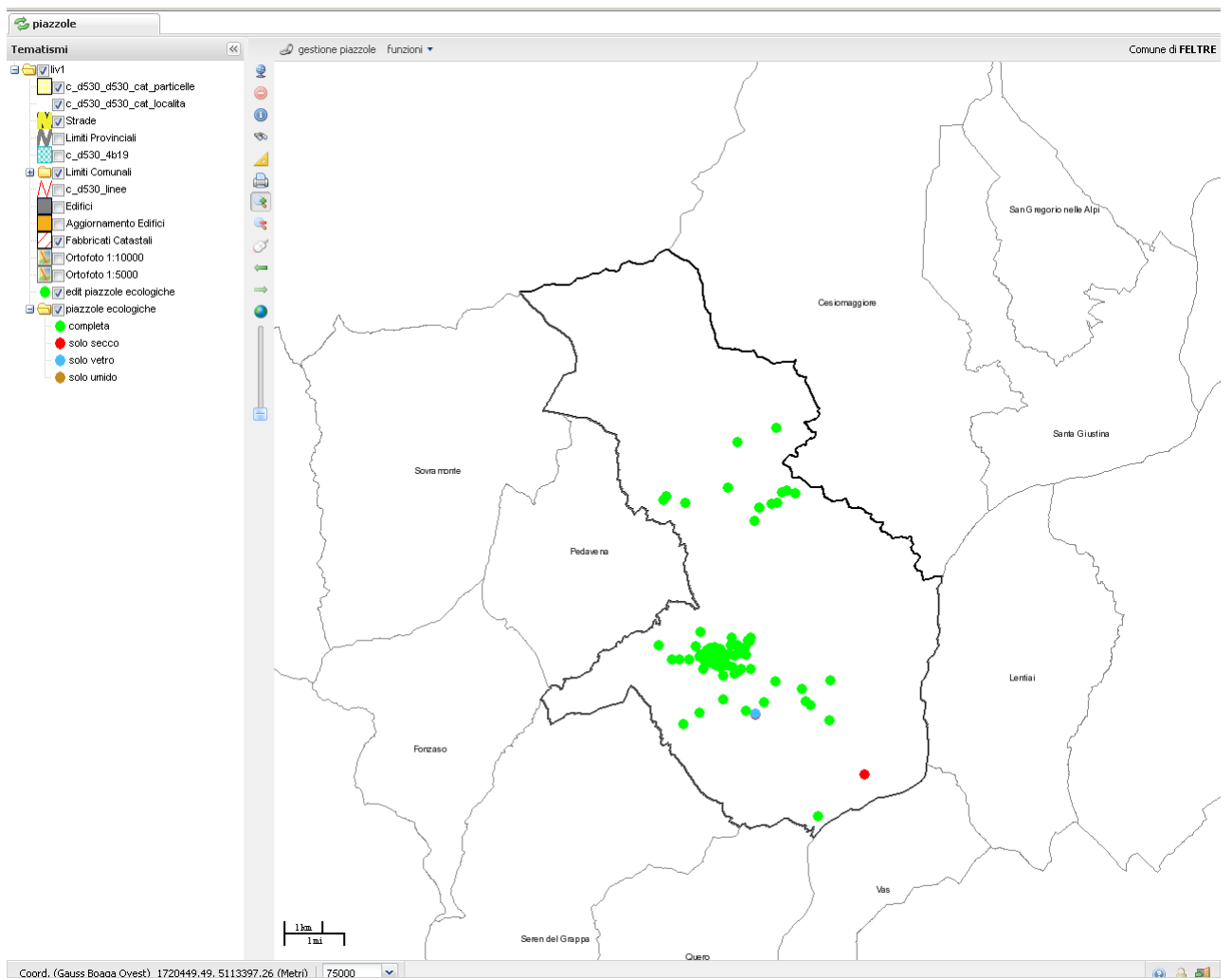
Queste ultime due entità possono sembrare superflue, in quanto basterebbe aggiungere alle due entità principali un attributo per indicare il tipo, ma è necessario memorizzare anche altri dati aggiuntivi (potrebbe essere necessarie anche modifiche in futuro).

5.2. DATI

5.2.1. CARTOGRAFIA

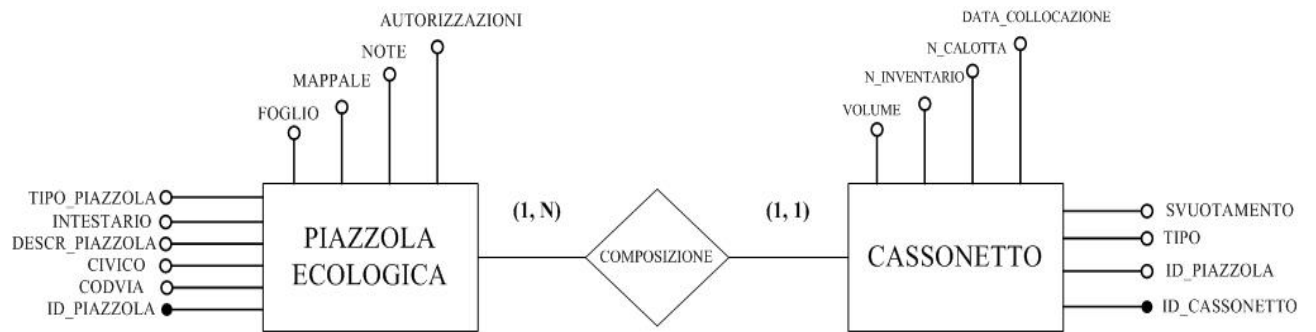
È stata realizzata la mappa per rappresentare tutte le piazzole dislocate sul territorio del Comune di Feltre. I layer cartografici (collezioni di oggetti geografici omogenei) inseriti sono i seguenti:

- Layer catastali: per mettono di visualizzare sulla mappa i principali dati catastali come foglio, particelle, edifici, nomi, limiti comunali.
- Layer per la visualizzazione delle strade.
- Ortofoto.
- Layer **edit_piazzole_ecologiche**: permette di inserire nella cartografia le piazzole ecologiche dislocate sul territorio comunale. Le piazzole vengono rappresentate con punti di colore verde.
- Layer **piazzole_ecologiche**: è collegato al tema precedente e anche al database gestionale. In particolare attivando questo tema è possibile visualizzare le piazzole già inserite nel layer precedente con colori diversi in base alla tipologia delle piazzole stesse (completa, solo secco, solo umido, solo vetro, etc.). Per realizzare ciò è stato utilizzato un campo di tipo numerico presente nella tabella che rappresenta le piazzole nel database gestionale. Per esempio al tipo di piazzola è stato associato il valore 0, al tipo solo secco il valore 1 e così via.

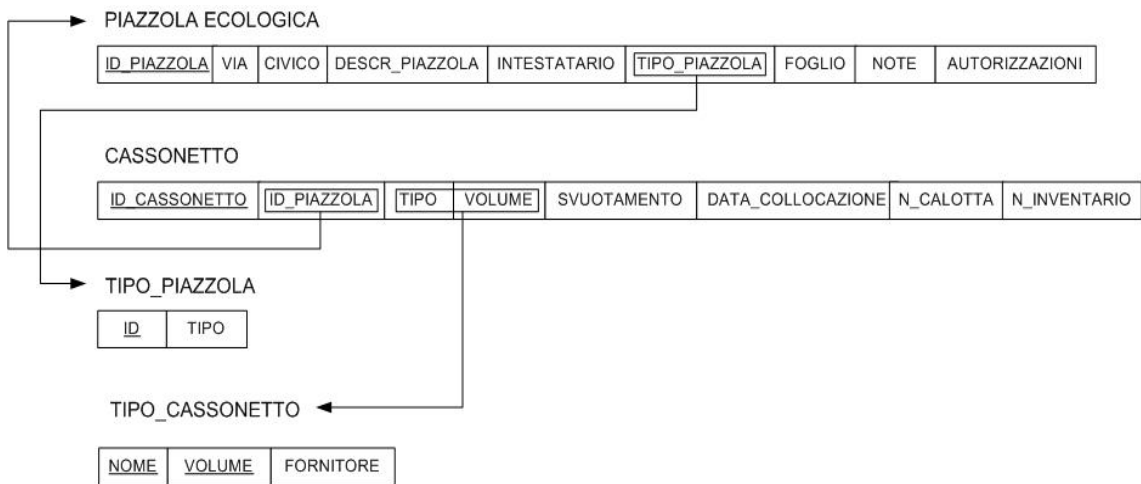


5.2.2. DATABASE

5.2.2.1. SCHEMA E-R



5.2.2.2. SCHEMA LOGICO



5.2.2.3. CODICE SQL

```
CREATE SEQUENCE cas_seq
    START WITH 1
    INCREMENT BY 1
    NO MAXVALUE
    NO MINVALUE
    CACHE 1;
```

```
CREATE TABLE contenitore
```

```
(
    nome character varying(20) NOT NULL,
    volume integer NOT NULL,
    fornitore character varying(50)
);
```

```
CREATE TABLE detail
```

```
(
    id_cassonetto      integer      DEFAULT      nextval
    ('cas_seq'::regclass) NOT NULL,
    id_piazzola integer NOT NULL,
    tipo character varying(20),
    svuotamento character varying(30),
    data_collocazione date,
    n_calotta character varying(10),
    n_inventario numeric,
    volume integer
);
```

```

CREATE TABLE master

(
    id_piazzola integer NOT NULL,
    codvia integer,
    civico integer,
    descr_piazzola text NOT NULL,
    note text,
    autorizzazioni text,
    foglio integer,
    mappale integer,
    intestatario character varying(50),
    tipo_piazzola integer
);

CREATE TABLE tipo_piazzola

(
    id integer PRIMARY KEY,
    tipo character varying(30)
);

CREATE SEQUENCE tipo_piazzola_id_seq
    START WITH 1
    INCREMENT BY 1
    NO MAXVALUE
    NO MINVALUE
    CACHE 1;

ALTER TABLE ONLY contenitore
    ADD CONSTRAINT contenitore_pkey PRIMARY KEY (nome,
volume);

```

```
ALTER TABLE ONLY detail
    ADD CONSTRAINT detail_pkey PRIMARY KEY (id_cassonetto);
```

```
ALTER TABLE ONLY master
    ADD CONSTRAINT master_pkey PRIMARY KEY (id_piazzola);
```

```
ALTER TABLE ONLY detail
    ADD CONSTRAINT "detail_ID_piazzola_fkey" FOREIGN KEY (
id_piazzola) REFERENCES master(id_piazzola);
```

```
ALTER TABLE ONLY master
    ADD CONSTRAINT "master_ID_piazzola_fkey" FOREIGN KEY (
tipo_piazzola) REFERENCES tipo_piazzola(id);
```

```
ALTER TABLE ONLY detail
    ADD CONSTRAINT "detail_tipo_contenitore" FOREIGN KEY (
tipo, volume) REFERENCES contenitore(tipo, volume);
```

5.3. APPLICATIVO WEB

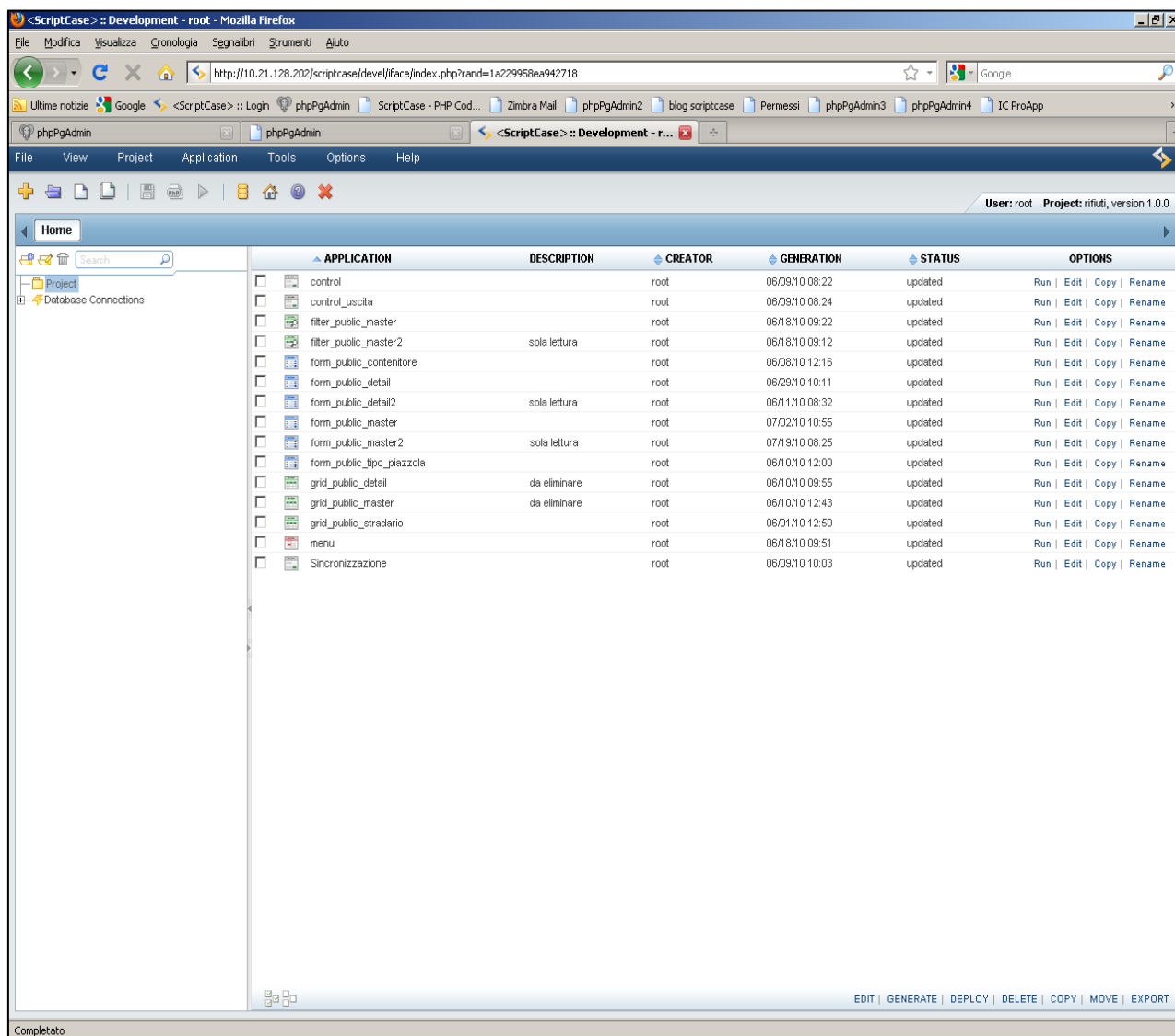
Per gestire i dati memorizzati nel database gestionale creato è stato deciso di sviluppare un applicativo WEB in PHP utilizzando lo strumento Scriptcase. L'applicazione creata viene poi pubblicata su uno dei server WEB del Comune e poi viene collegata al GIS.

L'ambiente di sviluppo Scriptcase permette, come già detto in precedenza, di produrre in maniera automatica codice PHP.

Utilizzando Scriptcase è possibile creare vari tipi di applicazioni:

- Form per l'inserimento e la visualizzazione dei dati
- Grid per la visualizzazione dei dati.
- Report.
- Menu.
- Applicazioni di controllo.
- Applicazioni per la ricerca dei dati.
- Applicazioni personalizzate.

Nella figura sotto è mostrata la schermata principale di Scriptcase con l'elenco di tutte le applicazioni create per il progetto Piazzole Ecologiche:

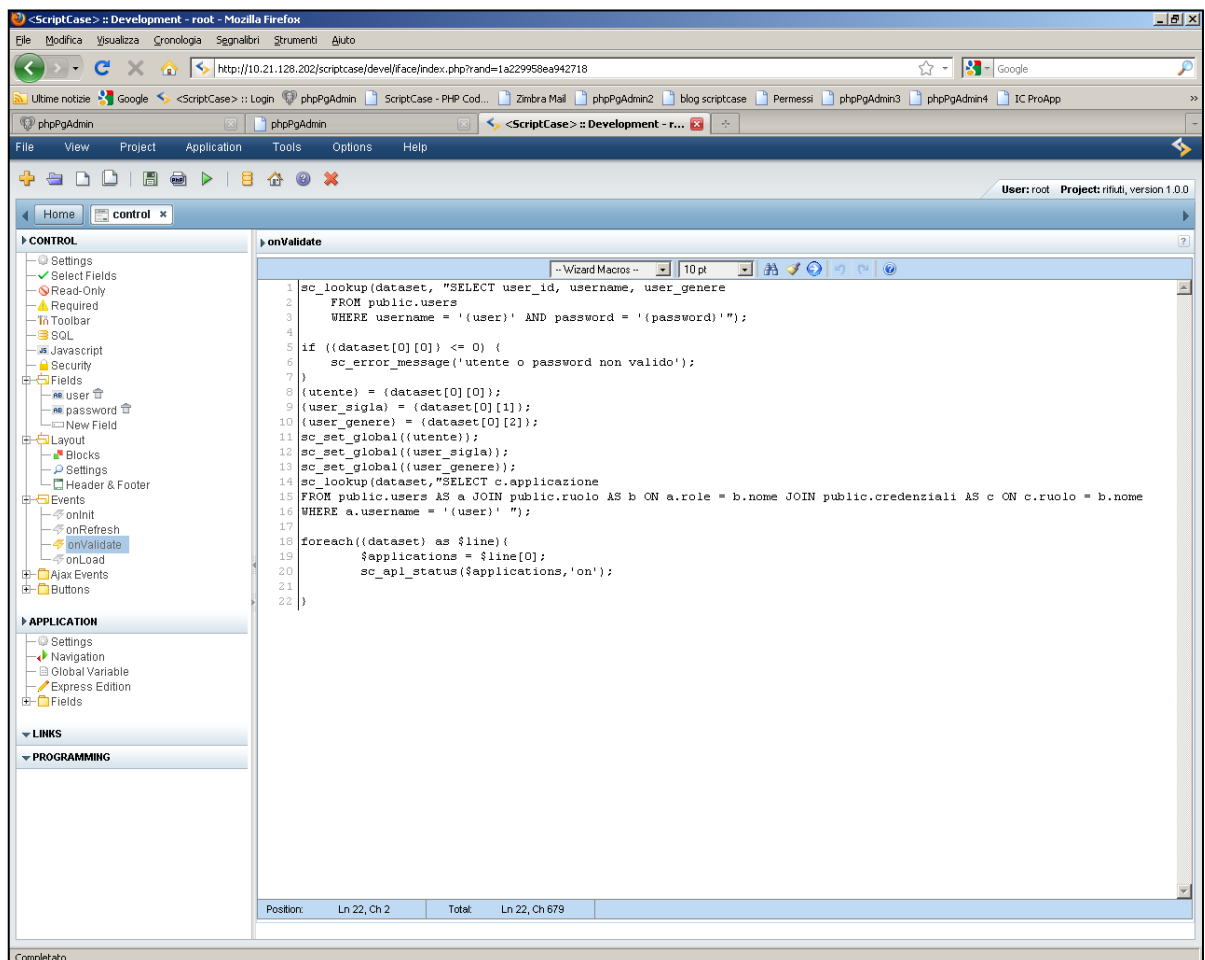


Come si può vedere, nella schermata principale sono presenti vari menù: File, View, Project, Application, Tools, Options e Help. Il menu Project permette di gestire tutti i parametri che riguardano l'intero progetto: creazione, eliminazione e la connessione al database su cui si basano le applicazioni. Il menù Application contiene le voci che consentono di creare, modificare, eseguire e pubblicare le applicazioni e in particolare contiene le voci “nuova applicazione”, “genera codice PHP”, “avvia applicazione” e “pubblica applicazione”.

Per il progetto in questione sono state realizzate 13 applicazioni, che verranno descritte di seguito:

- **Control:** si tratta di un applicazione di controllo che realizza il meccanismo di autenticazione per accedere al gestionale e si presenta quindi come la pagina iniziale del gestionale. Essa è composta da due campi, username e password e si collega al database Permessi, che gestisce i permessi e le credenziali degli utenti che devono accedere ai vari gestionali, per verificare che il nome utente e la password inserite siano corrette e, se questo controllo va a buon fine, vengono abilitate tutte le applicazioni del progetto per le quali l'utente possiede i permessi. In questo modo è possibile creare più utenti con credenziali e permessi differenti.

Nella figura sotto: il codice che realizza il controllo



- **Control_uscita:** realizza il logout (uscita) dall'applicativo e reindirizza l'utente alla pagina iniziale del gestionale.
- **Filter_public_master:** si tratta di un applicazione che realizza un meccanismo di ricerca di record di una tabella utilizzando uno o più attributi e in base a più criteri.

In questo caso è stata utilizzata per effettuare la ricerca dei record che rappresentano le piazzole ecologiche.

- **Grid_public_stradario:** quest'applicazione viene costruita sulla vista Stradario; viene utilizzata una vista perché la tabella Stradario originale può essere modificata solo dal personale comunale competente in materia. La grid creata visualizza tutti i record presenti nel vista Stradario che contiene tutti i dati relativi alle strade e alle vie del Comune di Feltre.
- **Menù:** realizza un menù che contiene le seguenti voci: Sincronizzazione, Inserimento, Ricerca e Esci, esse permettono in maniera veloce di accedere a tutte le applicazioni del progetto.

- **Sincronizzazione:** è l'applicazione che gestisce la sincronizzazione dei dati contenuti nel database cartografico con le informazioni memorizzate nel database gestionale. Essa accede al database cartografico, scorre tutti i record presenti e verifica se le piazzole sono presenti nel database gestionale, tramite il controllo sull'attributo 'piazzola_id'; questo campo è presente in entrambi i database, è un codice univoco (chiave primaria nel database gestionale, superchiave nel database cartografico) che identifica univocamente le piazzole. Infine se sono state inserite nuove piazzole nel database cartografico allora viene effettuato l'inserimento di quest'ultime nel gestionale.

In particolare il frammento di codice PHP che realizza ciò è il seguente:

```

sc_lookup (datigis, "SELECT piazzola_id,
    piazzola_descr FROM
    public.c_d530_piazzole_ecologiche", "postgis");
if (empty({datigis})) { sc_error_message('Nessuna
riga trovata.');
```

```

$count=0;
foreach({datigis} as $line)
{
    {id}=$line[0];
    {descr}=$line[1];
    sc_lookup (datigest, "SELECT id_piazzola,
    descr_piazzola FROM public.master WHERE
    id_piazzola={id}", "rifiuti");
    if (empty({datigest}))
    {
        sc_exec_sql ("INSERT INTO
        public.master(id_piazzola, descr_piazzola) VALUES
        ({id}, '{descr}'), "rifiuti");
        $count++;}}
{Record_inseriti}=$count;
print("Sono stati inseriti ".$count." record.");
```

Sono state utilizzate le funzioni PHP di Scriptcase:

- `Sc_lookup` (nome, comando SQL, connessione database): esegue un'interrogazione in codice SQL e memorizza il risultato in un array a cui viene dato come nome il primo parametro.
- `Sc_exec_sql`(comando SQL, connessione database): esegue semplicemente una query in codice SQL.

Per questo metodo sono state utilizzate, come già detto in precedenza, funzioni già fornite da Scriptcase ma si poteva ottenere lo stesso risultato utilizzando le funzioni fornite da PHP per l'interfacciamento con database PostgreSQL.

Per eseguire query SQL i passi da seguire sono i seguenti:

- Connessione al database:
`pg_connection($connection_string)` dove `connection_string` è una stringa che memorizza i dati della la connessione (host, database, utente, password).
- Esecuzione di una query:
`pg_query($connection, $query)` dove `connection` contiene i dati relativi alla connessione creata in precedenza e `query` è una stringa che contiene l'interrogazione di SQL.
- Recupero dei risultati della query:
`pg_fetch_array($result)` dove `result` è il risultato della query eseguita in precedenza.
- Chiusura della connessione:
`pg_close($connection)`.

- **Form_public_master:** è l'applicazione principale che permette di inserire i dati delle piazzole ecologiche. Viene costruita per interfacciarsi con la tabella principale, piazzole_ecologiche, che rappresenta appunto le piazzole.

Nella figura sotto: l'applicazione realizzata

- **Form_public_detail:** è l'applicazione realizzata per inserire i dati dei singoli cassonetti, che sono localizzati nelle varie piazzole, nella tabella corrispondente.

Nella figura sotto: l'applicazione realizzata

id_cassonetto	id_piazzola	tipo	volume	svuotamento	data_collocazione	n_inventario
1	1	Plastica	500			0
4	1	Plastica	500			0
3	1	Carta	800			0

- **Form_public_tipo_piazzola:** questa form permette di inserire nel database i vari tipi di piazzola previsti (completa, solo secco, etc.).

Nella figura sotto: l'applicazione realizzata



The screenshot shows a web application window titled '- public.tipo_piazzola' with a date '19/07/2010' in the top right corner. Below the title bar, there are four navigation buttons (back, forward, search, and another forward) and an 'Esci' button. The form contains two fields: 'id' with the value '1' and 'tipo' with the value 'completa'. At the bottom, there are three buttons: 'Nuovo', 'Aggiorna', and 'Elimina'.

- **Form_public_contenitore:** questa form permette di inserire nel database i vari tipi di contenitore previsti (vetro, carta, plastica, secco etc.).

Nella figura sotto: l'applicazione realizzata



The screenshot shows a web application window titled '- public.contenitore' with a date '19/07/2010' in the top right corner. Below the title bar, there are four navigation buttons (back, forward, search, and another forward) and an 'Esci' button. The form contains three fields: 'Nome' with the value 'Carta', 'Volume' with the value '400', and 'Fornitore' which is empty. At the bottom, there are three buttons: 'Nuovo', 'Aggiorna', and 'Elimina'.

6. CONSIDERAZIONI FINALI

I risultati ottenuti per il progetto isole ecologiche sono i seguenti:

- Il database gestionale per memorizzare tutti i dati relativi alle piazzole ecologiche per la raccolta dei rifiuti indifferenziati e per la raccolta differenziata.
- L'applicativo gestionale che permette di inserire e gestire i dati utilizzando un'interfaccia web.
- La mappa che permette di visualizzare la dislocazione delle isole ecologiche sul sistema GIS "InformCity".

Purtroppo durante il tirocinio non si è presentata l'occasione per interagire in maniera più attiva con il gis e i miei compiti si sono limitati alla progettazione database gestionali e sviluppo delle relative applicazioni web.